Optical Sensing of 3-D Contours for Online Control in Incremental Profile Forming

Undergraduate Honors Thesis

Presented in Partial Fulfillment of the Requirements for the Graduation with Honors Research Distinction in Mechanical Engineering at The Ohio State University

By

Ryo Nakahata

The Ohio State University

2019

Advisor: Dr. Srinivasan, Krishnaswamy

Abstract

Metal forming processes are facing demands for increased accuracy and geometric complexity of manufactured parts, even in small-lot production. However, metal forming processes are inherently subject to uncertainties in raw material characteristics as well as errors in the models of material behavior and of the manufacturing processes used for process planning. The impact of these uncertainties and model errors can be reduced through implementation of closed loop control of metal forming processes, based on online sensing of process variables of importance.

One of metal forming processes that benefit from such closed loop control is the incremental profile forming (IPF) process for tubular structures, recently developed at the Technical University of Dortmund, Germany. The desired geometry of the tubular cross-section is generated by indenters placed around the circumference, and commanded to execute the desired motions. Control of the indenter motions is not adequate to ensure the desired cross-sectional geometry after completion of the forming operation, since the latter depends also on the recovery of elastic deformation across the tubular structure following the removal of the forming load. Control of machine motions controls only the local geometry under the forming load.

For IPF process closed loop control, on-line sensing of profile geometry is therefore essential to improve the final part geometry. To develop robust closed loop control of IPF, non-contact sensing of part geometry is needed. This research evaluates the feasibility of optical sensing of 3-D contours on manufactured surfaces, and investigates the robustness of the measurement and the speed of data acquisition of the optical sensor for the IPF application. In addition, issues related to improving the part geometry based on optical sensing are considered.

A literature survey and determination of the sensing requirements for the IPF application was done, and indicated that the laser triangulation optical sensor best fulfilled the requirements. Profile geometry data on tubes was gathered using the sensor and issues related to sensor use under practical conditions of the IPF application were identified. Optical sensor data was compared with more accurate contact sensors, for the purpose of evaluating accuracy of the optical sensor. Sensor measurements were accurate to within 0.1mm, and data could be obtained at 40Hz. Sensing of the entire cross-section geometry requires multiple sensors that need to be used in a coordinated fashion, and issues in integrating the multiple sensor outputs were identified. Inaccuracy caused by the bending of the tube occurred during the IPF process continues to be a significant issue, which needs to be analyzed further. Future work related to the use of multiple sensors and options for eliminating the bending of the tube is also identified.

Acknowledgements

First, I would like to thank my advisor Dr. Srinivasan, Krishnaswamy for his help and support throughout this research project. He was always willing to helped me with by discussing about the research almost every week, even when I was in Germany doing the experiments.

I would also like to thank Professor Tekkaya and team of people at IUL for their support during the research internship at TU Dortmund.

I also would like to thank Juri Martschin and Eike Hoffmann for taking time and helping me with all machines I used in TU Dortmund.

Lastly, I would like to thank Dr. Grzancic for coming back to TU Dortmund to help my research as well as providing many of the data in this thesis.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Background on Incremental Profile Forming (IPF)

Metal forming is a technology that has been developing since the time of blacksmiths creating armors in the medieval ages. Even though metal forming has been developing for more than 7000 years, the demand for improvements in the metal forming process has increased significantly in recent years. One reason is that lightweight design of metallic structures and hence their manufacturing is becoming increasingly important factor in various industries. For example, within the automotive industry, requirements such as integrated functionality of parts and complex packaging constraints have increased the geometric complexity of profiles especially in tube forming. Increased geometric complexity leads to cross-sectional profile shapes varying along the tube's length.

Traditionally, methods such as hydroforming have been used in high volume manufacturing. Liquid under pressure is used to form the material into the desired shape using an appropriate die, as shown below in figure 1. However, this method requires an accurately machined mold, resulting in the forming process being inflexible in that it requires a new mold every time there is a slight design change in the final desired shape. Creating a new mold for prototyping and development is not suitable from both manufacturing time and cost in small-lot manufacturing.

In order to address the requirement for tubular structures in small-lot manufacturing applications, a metal forming process called incremental profile forming (IPF) was developed in 2013 at the Technical University of Dortmund. A picture of the IPF machine is shown below in figure 2. IPF is a metal forming process where multiple actuators profile the cross-sectional geometry of tubular metallic structures by controlling the radial motion of the actuators or

indenters. The tube is fed axially and further, the frame holding the indenters may be rotated as the tube is being fed, allowing the forming of helical profiles. The process thus relies upon the use of closed loop control of the multiple motions involved, to generate the desired tubular cross-sectional profiles [1].

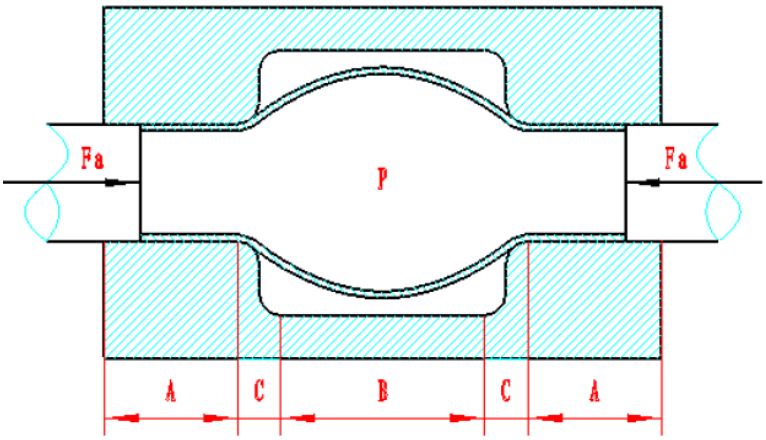

Figure 1: Schematic of tube hydroforming [2]

Figure 2: Picture of IPF machine [3]

The basic kinematics of IPF are shown in figure 3 below, where the multiple indenters on the machine, six in this case, are actuated electrically under servo control and create indents into

the tube with desired groove depths. A hydraulic actuator on the back of the machine moves the tube along the z direction as shown, under servo control. Rotation of the tube about the z-axis is the eighth degree of freedom, and is actuated electrically under servo control.

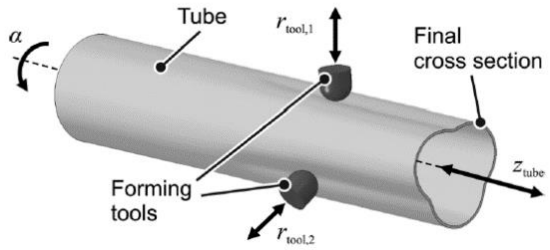

Figure 3: Principle of kinematic forming in IPF [1]

By relying upon kinematic control of the deformation process, the IPF machine solves the addresses the requirement of flexibility in small-lot manufacturing. For example, change in groove depths can be achieved just by changing the movement of the indenters appropriately. The multiple indenters in the IPF machine can be moved independently of each other, creating more freedom in how each of the tools can be placed in relation to the tube. In addition, the gravers or indenters may be replaced by support structures, either across from indenting tools or at other locations in the cross-section to stiffen the cross-section laterally. It is important to note that the flexibility of the process depends on the control of the eight degrees of freedom of motion substituting effectively for the reliance upon fixed and complex tooling inherent in processes such as hydroforming. Figure 4 shows an installation where all six tools are indenting the tube.

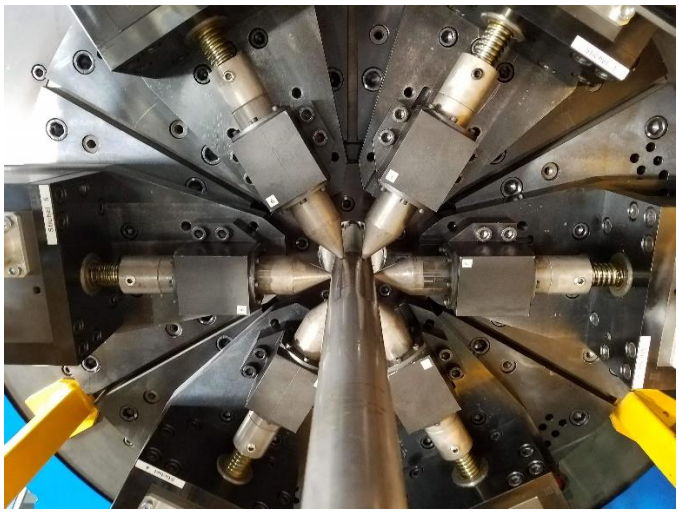

Figure 4: IPF machine while in the process of forming

The machine and process control configuration is shown in simplified form in figure 5. The process controller in the computer determines the machine motions needed to generate the desired geometry, and sends the reference displacements to servo control loops controlling the motions. Since multiple indenter motions may be involved, they are sensed individually and used as feedback to the multiple control loops controlling the indenter motions. The axial motion of the tube is also under servo control but the objectives of this control are simpler. The tube axial position is held constant during radial indentation processes. During axial grooving processes, the tube velocity is held constant at the prescribed level. Figure 5 does not show the provision for control of the rotational motion of the tool holders about the tube axis, for simplicity. This position is either held unchanged, for axial grooving operation, or varied at a constant rate for helical grooves. Servo control of this motion is also included in the machine control loops. All eight degrees of motion are sensed and fed back for control, as indicated in the figure.

It is important to note that servo control of the motions ensures only that the controlled motions are accurate at the time they are sensed, that is, during the forming operation and in the presence of the forming loads. Control of the motions does not guarantee that the geometry of

the tubular structure after the conclusion of the forming operation is accurate. This is emphasized in figure 5 since the indenter motions constitute only one set of inputs to the block labeled 'Process and Post-Process'. The additional input labeled 'Process and Post-Process disturbances' takes note of the fact that the groove geometry changes as the loading of the structure changes during the grooving operation and after the grooving operation when all forming loads are removed. The phenomena involved in this transformation certainly depend upon elastic recovery or springback, but also depend upon the changing nature of the loads on the tubular structure as the forming operation progresses along the tube. The underlying phenomena are quite complex and require good understanding of the process mechanics if they are to be modeled and incorporated in the overall control strategy.

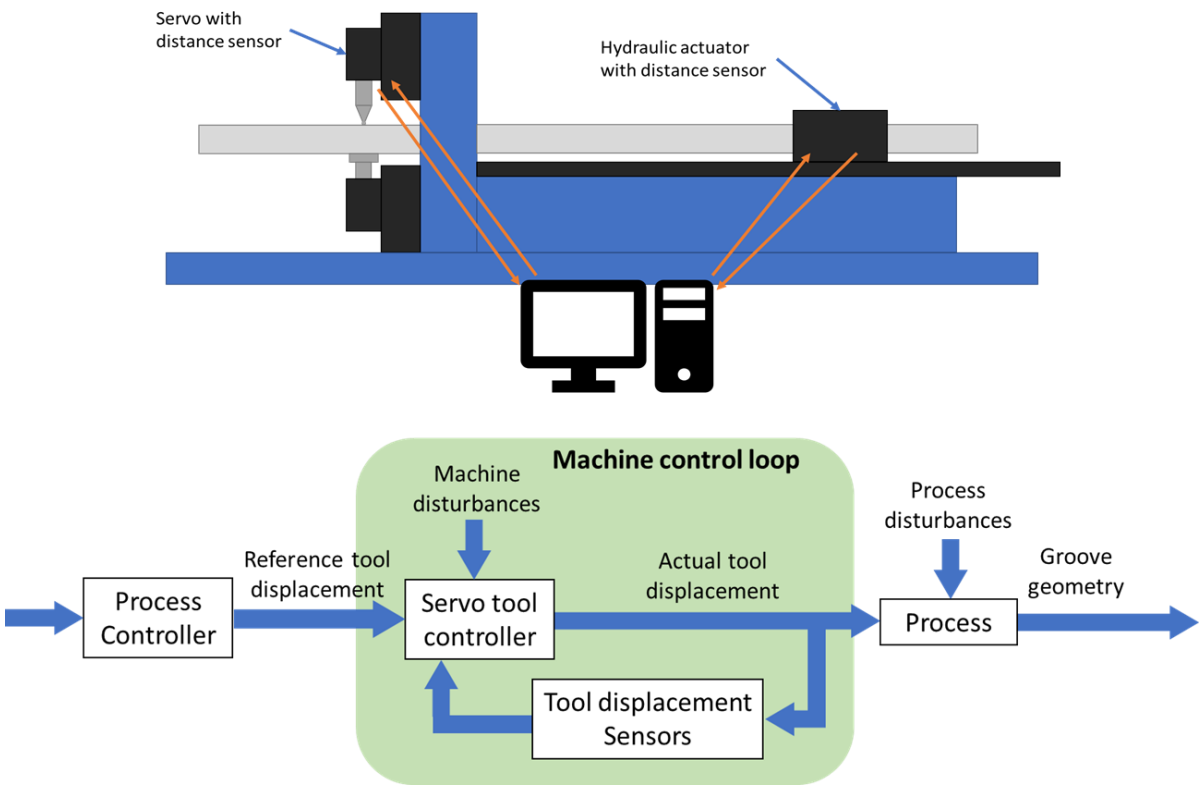

Figure 5: Simplified diagram of the IPF and control loops implemented in IPF machine

1.2 Limitations of the Incremental Profile Forming (IPF) Process

Incremental profile forming (IPF) cannot achieve geometrical accuracy of the formed structure, principally due to the fact that groove geometry is determined by a number of factors, only one of which is the accuracy of the servos controlling tool motions. As noted earlier, the evolution of the deforming loads on the structure during the forming process and elastic recovery of the structure during and after the forming operation affect the final and unloaded groove geometry. Material property variations from part to part, and material inhomogeneity, are other factors affecting final part geometry. If these disturbances affecting the final geometry are unknown and the process mechanics governing the final part geometry are imperfectly understood, the programming of the appropriate machine motions cannot accommodate these effects and the final part geometry will be in error. The importance of these factors for metal forming processes and their implications for the control of a variety of metal forming processes has been discussed at considerable length by Allwood et al [4].

Figure 6 shows the nature of the resulting groove geometry for two IPF processing strategies for generating a straight 100 mm long, 8 mm deep groove in a 40 mm diameter, 1 mm thick E235 steel tube using an indenter with a rigid hemispherical tool tip (4 mm radius). In both cases, the only sensors used for feedback are for machine control. The resulting unloaded groove geometry is measured after completion of the process on a Zeiss PRISMO coordinate measuring machine. The first processing strategy involves a single indenting pass with 8 mm depth, and the second involves eight indenting passes in the same direction, each of 1 mm depth. In both cases, the indenting tool performs a radial indentation to the desired depth at an indenting speed of 1 mm/sec, followed by axial grooving at a speed of 10 mm/sec, and then a radial tool withdrawal at 1 mm/sec speed. Clearly, in both cases, the resulting groove geometry is unacceptable, an

error of 0.1 mm or lower being the threshold. With the single-pass strategy, the groove geometry is non-uniform with the actual groove depth being more than 2 mm higher than the desired depth near the starting point, and lower than the desired depth by about 2 mm near the ending point of the groove. It is believed that this large non-uniformity in groove depth is a result of the very different loading and deformation patterns associated with the two processes of radial indentation and axial grooving for the high indentation depth per pass [2]. The multiple-pass strategy involving a smaller indentation depth per pass results in a nearly uniform groove depth along its length. The error in the groove depth is high, nearly 2 mm in this case. Also, the grooving time is eight times larger for the second strategy as compared to the first.

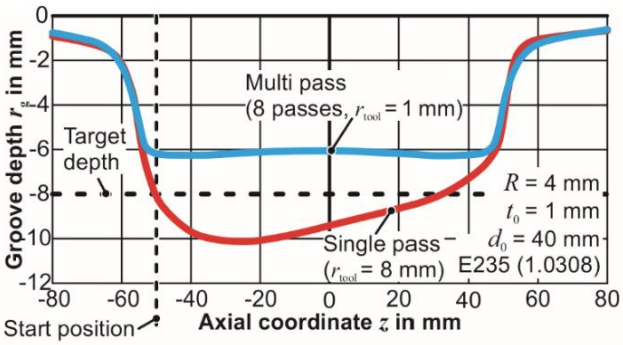

Figure 6: Measured groove depth using single and multi-pass IPF [3]

1.3 Focus and the Significance of the Research

One solution to improving the accuracy of control of any variable by closed loop control is to sense that variable online and adjust the control actions accordingly. Currently, the IPF sensing system consists of position sensors on the eight actuators powering the eight degrees of freedom of the machine [5]. Position sensors are simple to implement, and hence are used for IPF machine control. They perform the task they are intended for, namely, sensing of the corresponding actuator positions during the forming operation. They are clearly inadequate to ensure geometry of the resulting groove. The application of feedback control to improve groove

geometry accuracy must rely upon online sensing of the groove geometry, in addition to the sensing of the actuator motions.

Contact sensing of the groove geometry is one option but poses considerable challenges related to implementation since adaptation to profiles of variable geometry will require significant implementation effort for each profile. Clearly, this solution will run counter to the need for flexibility of the resulting process. Additionally, sensing groove geometry in process implies sensing groove geometry under load. This creates a problem since the final groove geometry can, and apparently does, differ significantly from that one would infer from the position sensors due to the factors stated above.

Noncontact sensing of the groove geometry would address one of the two problems noted above with contact sensing. The implementation effort needed for adapting the sensing scheme to a profile of specified geometry may be simplified if the optical sensing scheme is integrated with the actuation mechanisms used by IPF so that the sensing scheme need not be changed with each specific profile geometry. The issue of sensing the geometry under load would remain the same even with a noncontact sensing scheme. However, it is more likely that interruption of the IPF process can be done without impairing the quality of the resulting profile or increasing the grooving time significantly if noncontact sensing of groove geometry were done than if contact sensing were used. So, even though the answer to the latter question is unknown at this time, it seems that noncontact sensing of groove geometry is clearly the more profitable direction of investigation. Figure 7 shown below is the resulting control hierarchy of the IPF process that would result if sensing of the profile geometry is shown to be feasible. For closed loop control of the IPF process (outer loop control), sensing the groove geometry online is one of the main tasks, and is the focus of this research effort.

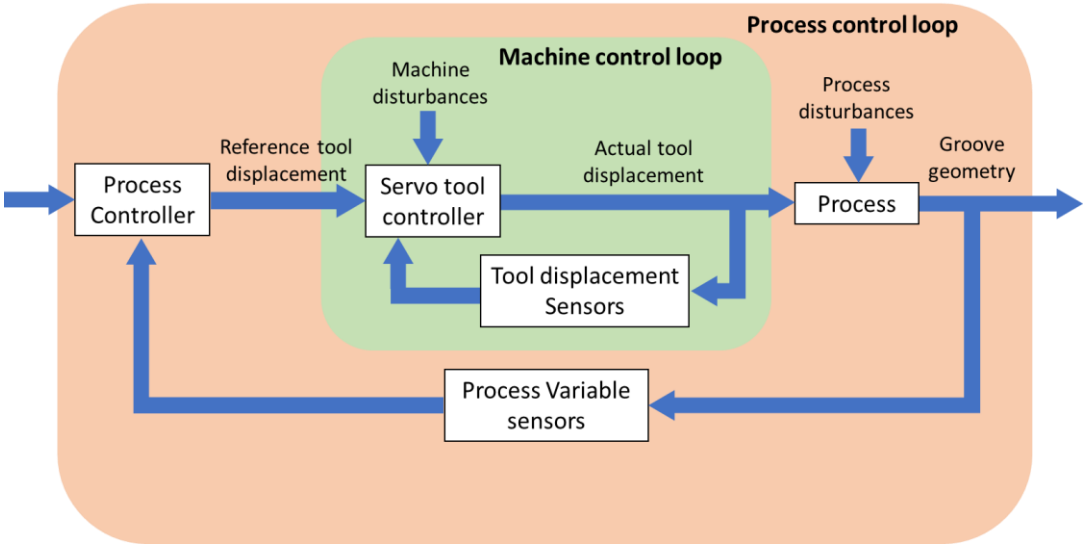

Figure 7: Control hierarchy for IPF process [5]

It is important to emphasize that sensing the groove geometry during grooving is only one of the issue that need to be addressed, to be able to come up with a control system that results in IPF processes that yield accurate profile geometries. The design of the process controller in figure 7 is another significant challenge. Specifically, the issue of design of the process controller in the figure requires the use of process knowledge to determine how the machine control reference tool displacements need to be adjusted during the grooving process, based on the sensed groove geometry.

The underlying challenge is not inability to model the process mechanics with sufficient accuracy. In fact, the process has been modeled using finite element models (FEM) and simulated to obtain predicted geometry results similar to those obtained experimentally as documented extensively by Grzancic [2]. However, creating and running the FE model requires significant computational power and time. For example, the computations for a single axial groove took 24 hours when using three Intel Xeon cores [5]. The challenge is in reducing the process knowledge to a form that allows both model-based controller design and online modification of the machine control reference tool displacements. For a simple groove of

constant desired depth, such as that in figure 6, such a path would start with a smaller initial indenter depth and increase the indenter depth as the tool travels axially. Such a general shape of the indenting tool path has been shown to result in improved groove geometry for a simple straight groove [5]. However, for an IPF process and part geometry of any realistic complexity, determination of the nominal indenter tool paths would require considerable effort as the problem being solved is the inversion of the FE model of a geometrically complex, nonlinear, time varying forming process. Controller design for the control structure in figure 7 is thus a larger research challenge in its own right and is outside the scope of the current research effort.

In order to sense the groove geometry for the process control loop, non-contact optical sensing would seem to be a promising candidate for online geometry measurement and feedback for closed loop control. The immediate goals of current research are to evaluate the feasibility of optical sensing of 3-D contours on manufactured surfaces through a literature survey, select the sensor best suited for the IPF application, determine the online sensing requirements of the IPF application, and determine any limitations of the selected sensor based upon experimental evaluation. Other goals of the research relate to subsequent use of the chosen optical sensor for control of the IPF process. These include obtaining deeper understanding of how to utilize the information gathered from the optical sensors in conjunction with process models to better control the process. The implications of sensing the groove under partial loading are identified. Further, the requirements on the sensors for sensing the entire tube cross-section are identified. The use of the sensing scheme to address other requirements of the application are also investigated. In particular, preliminary considerations in use of the sensing scheme for correcting tube orientation prior to the grooving are determined. Finally, some of the lessons learnt in

selecting and applying optical sensing to the IPF process are expected to benefit other metal forming processes.

1.4 Overview of the Thesis

This thesis includes 5 chapters. Chapter 2 discusses the determination of optical sensor that is suitable for control of the IPF process. The first part of the chapter discusses different possible sensors that can be used, through literature review. Sensing options available at the Technical University of Dortmund are evaluated next, including examination of a laser triangulation sensor and the 3D ATOS Triple Scan System. Chapter 3 discusses different sensor properties of the selected sensor through literature review and experimental evaluation of the sensors. Specifically, this chapter discusses accuracy and repeatability of the sensor, as well as the data processing needed to gather both cross-sectional and longitudinal contours of the groove. This discussion includes the LabVIEW program used to collect sensor data and analysis using MATLAB. This chapter also discusses the simulated use of multiple sensors to determine the feasibility of sensing the entire cross-sectional contour of the deformed tube. Chapter 4 discusses the process insights from sensing of the contours, especially how the different tool geometry affects the final tube geometry. Chapter 5, the concluding chapter, summarizes the findings from the research, and recommendations for future work. This chapter also discusses preliminary considerations related to future work, including development of a sensor mount for the multiple sensors required for gathering the entire cross-sectional contour, and creation of a tube orientation control system using such feedback, prior to the forming operation.

Chapter 2: Selection of the Optical Sensor

2.1 Literature Review

For this initial exercise in selection of the optical sensor, four different optical sensing candidates were considered and their properties compared with each other to determine the sensor that is best suited to measure cross-sectional geometry of the tubes formed by the IPF process. The four candidates – a laser triangulation sensor, a confocal sensor, a speed of light sensor, and a 3D point cloud data sensor were compared in terms of their specifications and their uses. All the chosen sensors were noncontact sensors, and did not involve tactile sensing.

Laser triangulation is a method whereby a laser is projected onto the sample that is being measured, and a CCD (Charge Coupled Display) camera determines the depth of the sample through the position of the projected laser, as shown in figure 8. There are multiple reported cases of this sensor being used for metal forming applications. One example involved use of multiple line lasers and CCD cameras to measure roll formed U-shaped profiles for use in closed loop control [6], figure 9. One of the limitations of this sensor is that the background of the raw image obtained from the CCD camera is noisy [7]. Thus, this method requires image processing in order for the image data to be usable.

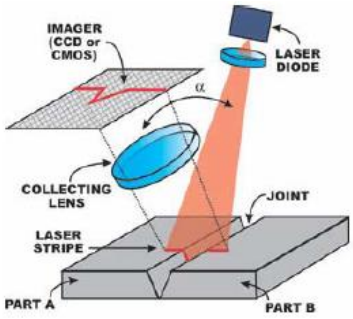

Figure 8: Laser triangulation sensing principle [7]

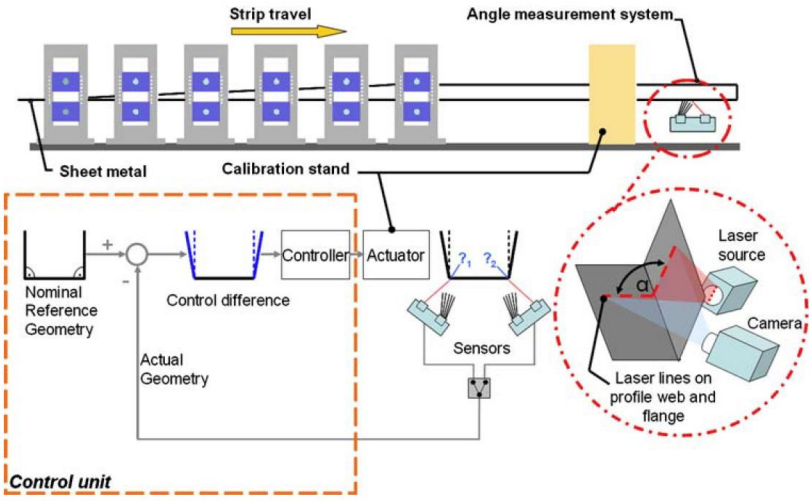

Figure 9: Laser triangulation sensor implementation in control loop of U-shaped metal sheet forming [6]

A confocal sensor senses the reflection of a polychromatic beam to determine the distance of an object. Different wavelengths are focused at different points to enable the sensor to gather accurate information about the distance of the object [8]. This sensor is ideal for conditions where fast distance or thickness measurement is required. One example involved use of confocal sensor to determine the fill level of microtiters as shown in figure 10. However, due to the manner in which the sensor senses the color of the reflected light, the object needs to be exactly perpendicular to the direction of the light in order to determine an accurate distance. Figure 11 below shows the principal components and the principles involved in confocal sensing.

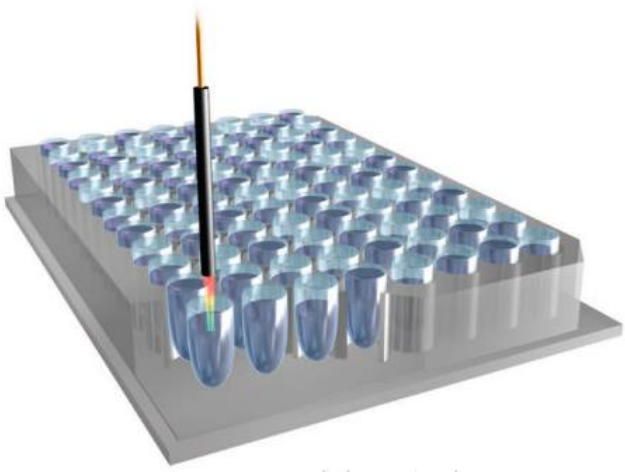

Figure 10: One example use case of confocal sensor [8]

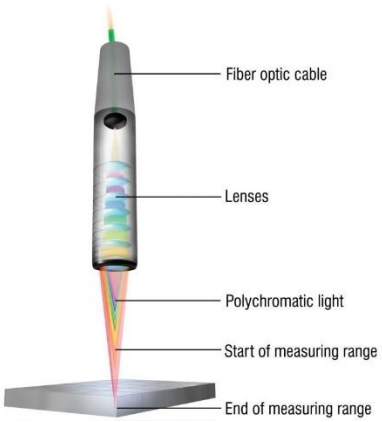

Figure 11 Principles of confocal sensing [8]

A speed of light sensor is a single point sensor that can measure distance by calculating the time taken for light to travel to the sensed object and then back to the sensor [9]. This sensor is usually used for positioning and type classification in machine building and handling equipment. Also, this sensor has high repeatability as well as very fast response due to its robustness. However, this sensor is a single point sensor, similar to the confocal sensor, and therefore requires for the IPF application that either the sensor or the tube rotate in order to capture the whole cross-sectional contour.

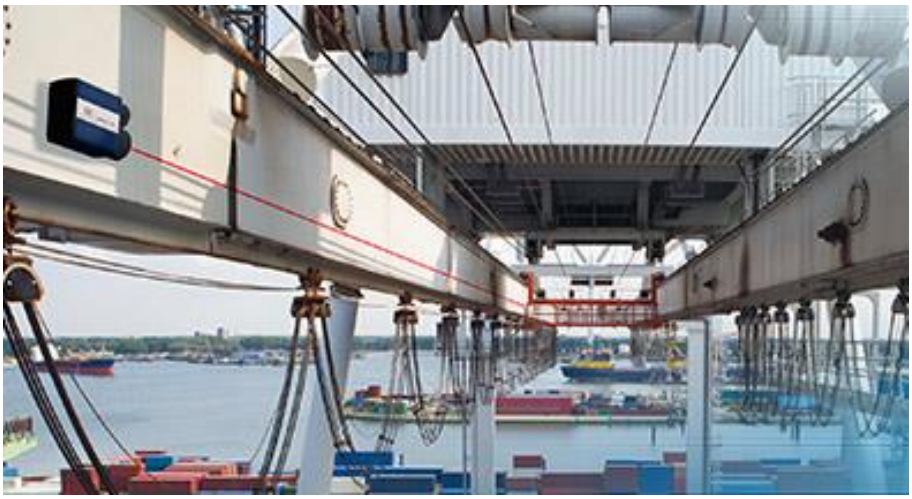

Figure 12: One example use case of speed of light sensor [9]

Lastly, 3-D point cloud data sensing uses the stereo camera principle, where multiple point lasers create a constant pattern and the camera captures the image and juxtaposes the image on a reference image, and compute the distances involved. One of the commercial sensors that uses this method is the ATOS Triple Scan sensor (Figure 13), where precise patterns are projected onto the surface to be sensed and their images captured by two cameras [10]. One example use case of ATOS Triple Scan system is determining minor defects in car door made from metal forming as shown in figure 14. However, one of the problems with this sensor is that random error of depth measurements increase with increasing distance to the sensor [11]. Thus, this error must be considered when using an optical sensor utilizing 3-D point cloud data.

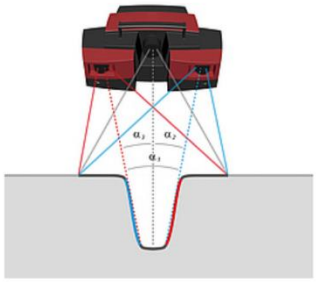

Figure 13: ATOS Triple Scan system [10]

Figure 14: One example use case of ATOS Triple Scan system [10]

Since there are many different sensors employing the different sensing principles, one sensor employing each of the different sensing schemes was identified, and used for comparison. Table 1 below shows the sensors selected thus. The different sensors were compared by using sensor specifications noted by the manufactures. Tables 2 and 3 below show different quantitative and qualitative properties of the sensors listed in table 1.

For the laser triangulation sensor, the accuracy was not as great as confocal sensors, with the laser triangulation sensor having a 4 μm resolution compared to the confocal sensor which has 180 nm resolution. However, the confocal sensor requires angle adjustment on the material, which this will not be possible due to the fact that the tubular cross-section in IPF is not flat. Thus, implementation of the confocal sensor is much more difficult than laser triangulation sensor. Also, the 4 μm resolution of the laser triangulation sensor is adequate for the IPF application.

The speed of light sensor is a single point sensor that can measure distance from time taken. This is a robust system, but the one of the most accurate speed of light sensors could only have a resolution of 0.1mm, with linearity error of 5mm, which is not suitable for measuring the contour of the grooves in IPF.

Table 1: Sensors selected for comparison

Sensing Method	Sensor Selected
Laser triangulation	Micro-Epsilon scanCONTROL 2910-50/BL
Confocal sensing	Micro-Epsilon IFS 2405-30
Speed of light	Micro-Epsilon ILR 1182-30
3D point cloud data	GOM ATOS Triple Scan System

Table 2: Quantitative Properties of different sensors

Sensor type	Resolution (µm)	Linearity error (µm)	Frequency (Hz)	Depth of Measurement (mm)	Data points per profile
Laser triangulation	4	50	300 (variable)	50	1280
Confocal	0.18	7.5	100-70000	30	1
Speed of light	100	5000	50	17000	1
3D point cloud data	27	N/A*	N/A*	300	N/A*

(*Values are either unknown or variable depending on the surface quality and object placement)

For 3D point cloud data, the difficulty in calibration and low mobility was a key problem that could become a major problem in using the sensor as a feedback sensing system. However, due to how movement required for obtaining the whole 3D contour is only capturing 3 to 4 profiles, it would be easier to mount the sensor onto the IPF machine compared with sensors that require rotation of 360 degrees around the object.

Figure 15 below shows a web-graph that shows a visual comparison of the different sensors. The corresponding reference score representation is shown in table 4, where a score of '1' means poor and a score of '5' means good.

Table 3: Qualitative Properties of different sensors

Sensor type	Movement required for capturing 3D contour	Ease of setting up	Flexibility/ Mobility	Effect on the angle of the sensed surface
Laser triangulation	Capture 3 or 4 profiles and z movement	Quick and has LabVIEW interface	Compact and easy	Low
Confocal	Rotation of 360 degrees and z movement	Difficult calibration but has LabVIEW interface	Compact and easy	High
Speed of light	Rotation of 360 degrees and z movement	Quick and has LabVIEW interface	Compact and easy	High
3D point cloud data	Capturing 3 or 4 profiles	Difficult calibration and no LabVIEW interface	Large and difficult to move around	Low

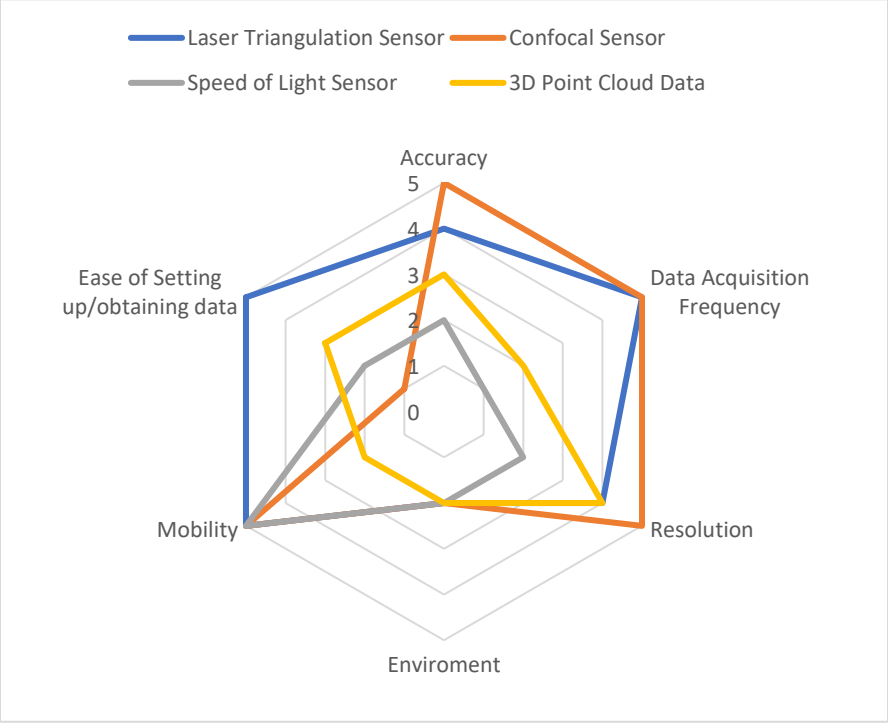

Figure 15: Web-graph of different sensors

Table 4: Score reference for the web-graph

Sensor Properties	1 (poor)	5 (good)
Accuracy	Low accuracy	High Accuracy
Data Acquisition Frequency	Low Frequency	High Frequency
Resolution	Low Resolution	High Resolution
Environment	Greatly affected by environment	Not affected by environment
Mobility	Difficult/impossible to move around	Easy to move around
Ease of Setting up	Difficult calibration and no LabVIEW interface	Has LabVIEW interface and easy calibration

As can be seen on the web-graph, the laser triangulation sensor seems to have the best overall score while others scored low on ease of set up and data acquisition. The confocal sensor scored high on resolution, data acquisition frequency, accuracy, and mobility, but scored the lowest on ease of set up and data acquisition. The requirement that the sensor needs to be perpendicular to the object for accuracy is responsible for the poor score. In addition, since the sensor provides only a point measurement, sensor movement of 360 degrees around the object is required, making the setup very difficult.

2.2 Examination of Sensing Options at TU Dortmund

At the Technical University of Dortmund, a laser triangulation sensor and a 3D point cloud data sensor were available to examine. The model that TU Dortmund had for laser triangulation sensing was Micro-Epsilon scanCONTROL 2910-50/BL, which emits blue line laser to determine the geometry. The sensor was easily implemented in LabVIEW and was easy to set up. However, due to the amount of degree of freedom that the sensor had, there was a need to restrict sensor motion in all 6 degrees of freedom, which can be very difficult to do.

For 3D point cloud data sensing, the GOM ATOS Triple Scan System was the model that TU Dortmund had. Implementation of 3D point cloud data for sensing of IPF geometry was considered, but even though the system uses 3D data with many data points, it has a very

difficult calibration procedure and was not accurate enough. This is due to the fact that 3D point cloud data requires reference points and the area for which data is gathered in this method is much greater than for any other optical sensor listed. Thus, 3D point cloud data sensors can be better used for offline measurements of the formed tube, but is not suitable for online measurements.

After looking into many different options of optical sensors, the laser triangulation sensor was selected to be tested for measuring the 3D-contour of the manufactured part in IPF. The selected sensor was easily controllable with LabVIEW software, and the sampling frequency was fast enough that it should not affect the IPF machine's control frequency of 40Hz.

Chapter 3: Experimental Evaluation of Laser Triangulation Sensing of Geometry in IPF Processes

3.1 Properties of the sensor

The laser triangulation sensor, Micro-Epsilon scanCONTROL Model 2910-50/BL, was analyzed further for a deeper understanding of its properties and capabilities. As mentioned in the past chapter, the sensor was stated to have a 300 Hz refresh rate for capturing the profile. This limitation arises from limitations of processing hardware in the device [12]. Shown below are the dimensions of the sensor used for experiments on the IPF machine.

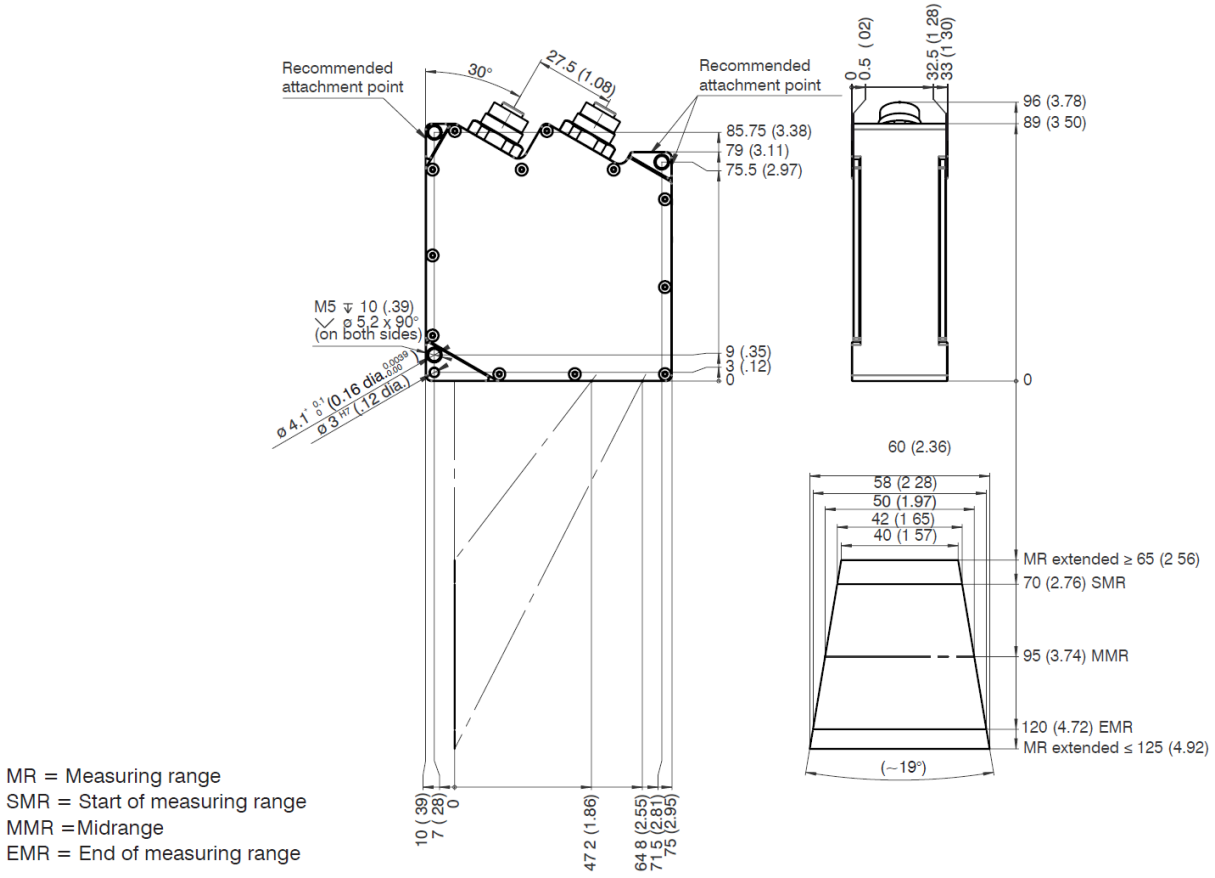

Figure 16: Sensor dimensions and field of view [12]

The dimensions of the sensor are 85mm long, 33mm wide, and around 90mm tall if the cable connectors are disregarded. The laser stripe is oriented vertically 10mm from the left as shown in the figure above. As can be seen, the laser triangulation sensor has a low profile, which means it can be easily integrated into the IPF machine as needed, a trait that makes use of the sensor for a variety of IPF profiles easy without additional setup. The measuring range is shown above very clearly in the bottom right corner of the figure, with the start of the measuring range 70 mm away from the sensor up to the end of the measuring range of 120 mm. This means that the tube contour needs to be always within that range in order to be scannable with the laser triangulation sensor. There is also a horizontal measuring range shown in the figure above, where the minimum horizontal range of 48 mm occurs at the vertical distance of 70 mm and the maximum horizontal range of 58 mm occurs at the vertical distance of 120 mm.

The laser triangulation sensor uses a LabVIEW interface for data communication between the sensor and the computer. The LabVIEW function to determine the profile was provided by Micro-Epsilon. However, the functions needed to save a continuous profile were not given and were developed in LabVIEW code. Communication between the sensor and the computer was done through using the ethernet port, which can transmit data at the rate of 100 Mbit, but not power for the sensor.

There are some limitations that the manufacturer states in the quick reference guide. According to the manufacturer, the reflection of the target surface can have an effect on the accuracy of the measurement, if the direct reflection does not reflect all the light into the camera, making the field of view narrower. Thus, the manufacturer recommends doing a preliminary examination of the surface of the target. Other factors that can affect the measurement accuracy are the color differences in the measurement objects. Since the exposure parameters can only be

changed as a whole for one profile, the color difference in a single scan can make some parts of the profile to be too exposed or obscure some parts of the profile. Other minor factors include intensity of the external light, mechanical vibration, and surface roughness if resolution in the micrometer range is required. Since the IPF application did not require this high a resolution at present, this limitation was not a factor in the current work.

3.2 Accuracy and repeatability of the sensor for a single scan

In order to determine the accuracy of the sensor for a single scan, the profile geometry gathered by the optical sensor was compared with the Zeiss PRISMO System which is a coordinate measurement machine using a contact sensor. The picture of the Zeiss is shown below in figure 17 for reference.

Figure 17: Picture of the Zeiss PRISMO System used for evaluation of accuracy

The Zeiss system was used as a reference against which to evaluate the optical sensor since its accuracy was between 0.4 µm and 1.7 µm depending on the tool tip used. This resolution, at the high end, is almost 10 times smaller than the resolution of the laser

triangulation sensor. Hence, the accuracy of the Zeiss is adequate for evaluating the accuracy of the laser triangulation sensor.

The first experiment that was done was to first measure the same groove geometry with the laser triangulation sensor if the surface emissivity was altered. This was done using a white chalk to coat the groove and comparing the result with that obtained without such a coating. For this experiment, ambient light is used and the distance between the target and the sensor is kept unchanged. The tubes used for this experiment are: Steel tube (E235) with a single radial indentation, Aluminum tube (AL6060) with a single axial groove, and a Steel tube (E235) with a flat groove. These grooves were created more than 1 month prior to the measurement, thus eliminating any post-process phenomena occurring between scans with the laser triangulation sensor and the Zeiss.

Figures 18-20 show the different targets used for determining the accuracy of the laser triangulation sensor on metallic surfaces. As may be seen in the figures, all three surfaces have different geometry as well as different levels of reflection at the measurement locations, making the results from this experiment generally useful for a variety of IPF processes and part geometries. In order to test the effect of a change in the surface emissivity, the different surfaces were coated with white chalk, an example of a tube coated thus being shown figure 19. Thus, there were six different measurements done by the laser triangulation sensor, and those six measurements were compared with the measurements done by the Zeiss on the three contours.

Figure 18: Picture of the steel tube with a single radial indentation

Figure 19: Picture of the aluminum tube with a single axial groove

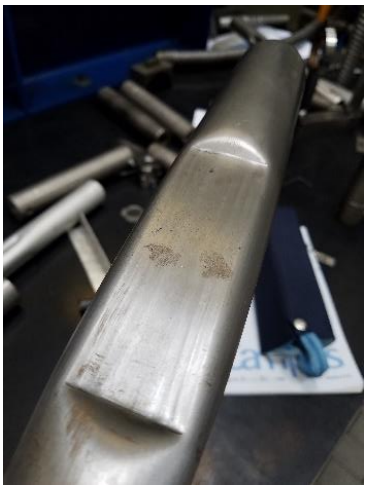

Figure 20: Picture of the steel tube with a flat groove

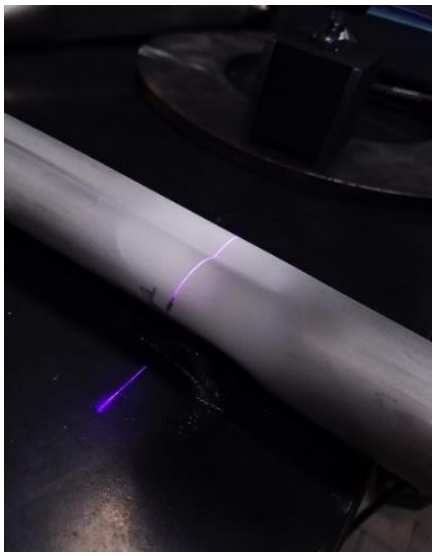

Figure 21: Picture of the aluminum tube with white chalk

Figure 20 shows the setup was used for this experiment, with the sensor being placed some distance away from the table and the tube placed flat on the table. The measurement line was recorded by a black marker to indicate the exact location of the measurements on the tubes. As shown in the figure, the sensor was attached to the sensor support, which was attached rigidly to the table to provide support stability for all of the measurements done by the laser triangulation sensor. The data obtained from the sensor was analyzed in MATLAB to determine

both the accuracy of the sensor, as well as how surface emissivity affected the sensor reading and its accuracy.

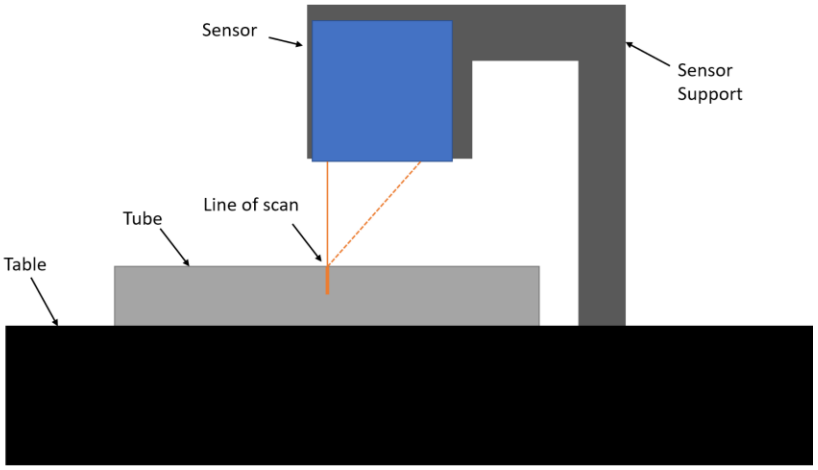

Figure 22: Diagram of the test setup for sensor accuracy testing

Comparison of the data obtained by the triangulation sensor and the Zeiss for the aluminum tube is shown in figure 23, the remaining two comparison plots being included in the Appendix. As may be seen from the plots, displacements measured for both the raw metallic surface and the tube with white chalk display the same displacement traces as the measurements done by the Zeiss, based on a visual inspection. Figure 23 indicates that the three traces agree with each other to within 0.1 mm, suggesting that the effect of surface emissivity changes is under this limit.

Since the profile gathered by the laser triangulation sensor follows the Zeiss measurement very closely, the deviation between the two types of sensor measurements was studied more closely. Figures 24 – 26 show that the deviation between the two sets of measurements is fairly small, between -0.08mm to 0.08mm over most of the measurement region. As noted in figure 24, the deviation for the single radial indentation is not as accurate in the - 10 mm to – 4 mm range. While all three measured contours had a deviation within 0.17 mm over the complete range of

interest, the deviation was between $\pm$ 0.08 mm over most of the measurements. In terms of the effect of emissivity, it seems that all readings for the raw metallic surface profile seem to have slightly more inconsistency in the readings close to each other, while readings for surfaces coated with white chalk seem to be more consistent. However, these inconsistencies are minimal, and it can be concluded that surface emissivity has minimal effect on laser triangulation sensor measurements.

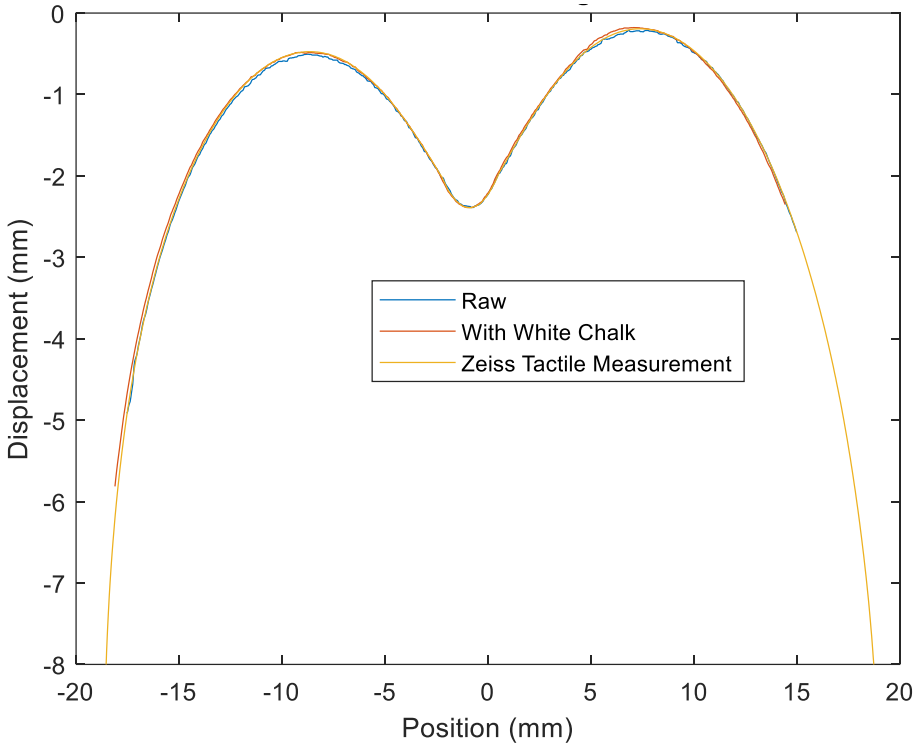

Figure 23: Sensor accuracy comparison for aluminum tube with single groove

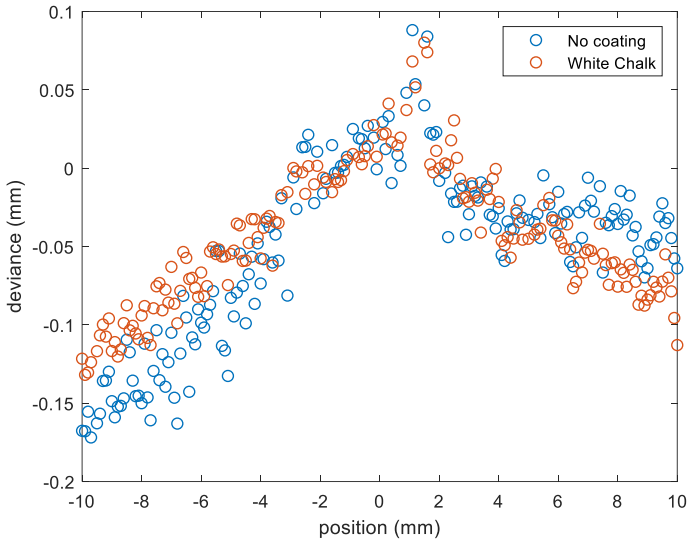

Figure 24: Deviation of laser triangulation data with surface emissivity (steel radial indent)

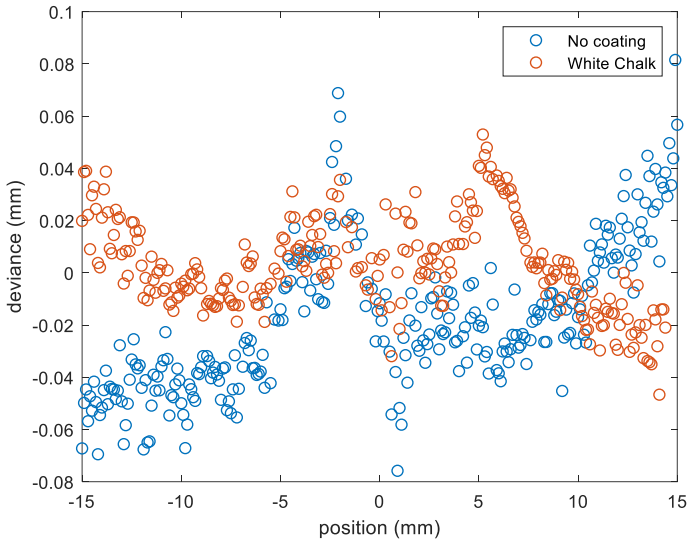

Figure 25: Deviation of laser triangulation data (axial groove in aluminum)

3.3 Process to gather the cross-sectional contour

In order to determine the cross-sectional contour of the tube while it is in the IPF machine, the laser triangulation sensor need to be mounted with stability inside the IPF machine frame. Initially, magnet mount with a 5 degree-of-freedom arm was used for the laser mounting.

However, this resulted in poor stability, and there was a need for creating a more stable mount for the sensor on the IPF machine. The original magnet mount is shown in figure 27.

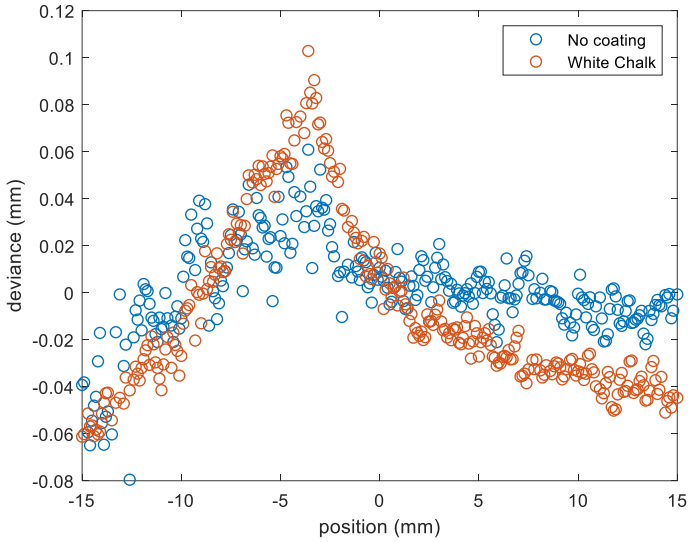

Figure 26: Deviation of the laser triangulation data (Steel flat groove)

Figure 27: Magnet mount on the IPF machine with 5 degree-of-freedom arm

The new mounting system required a mechanism for some adjustment while retaining stability. In order to meet these requirements, a 3D model of the new mount was created through the use of SolidWorks. The part was created through a 3D printed bracket, as well as sheet metal with different thickness, with the sheet metals being cut into desired shapes using a steel sheet

laser cutter. The SolidWorks model for this mount assembly is shown in figure 28, along with the assembled final support.

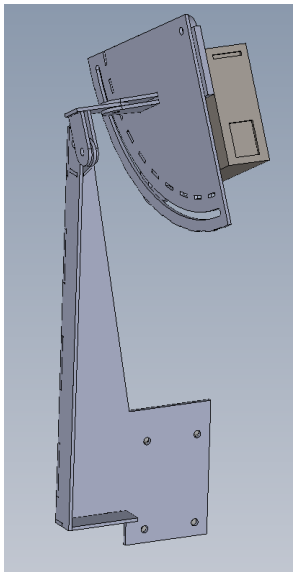

Figure 28: 3D model of the new mounting system

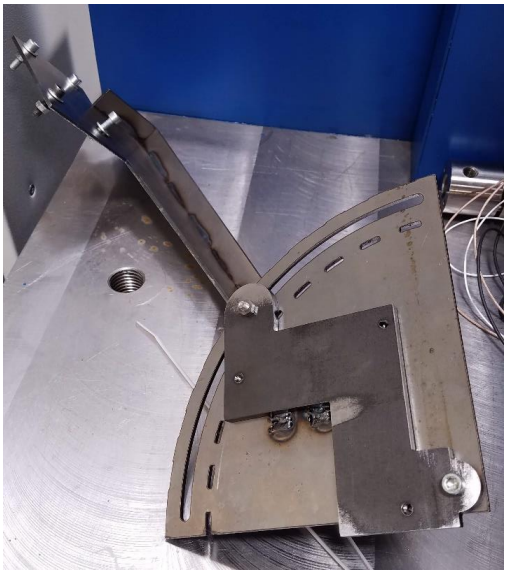

Figure 29: Assembled sensor mount

The assembled sensor mount was then mounted on the IPF machine to be used for continuous sensing of part geometry. The mount was successful in holding the sensor with stability and was able to sense the cross-sectional contour of the tube. The picture of the sensor mount being installed on the IPF machine is shown in figure 30. As may be seen in the figure, the sensor shines laser light on the groove, enabling easy identification of the groove on the tube as the groove or indent is being formed.

Figure 31 shows the raw groove data overlaid with the original contour, both measurements being obtained using the laser triangulation sensor on custom sensor mount. As may be seen in the figure, the data captures the cross-section of the indent that was created in the tube and also shows how deep the groove is at the center.

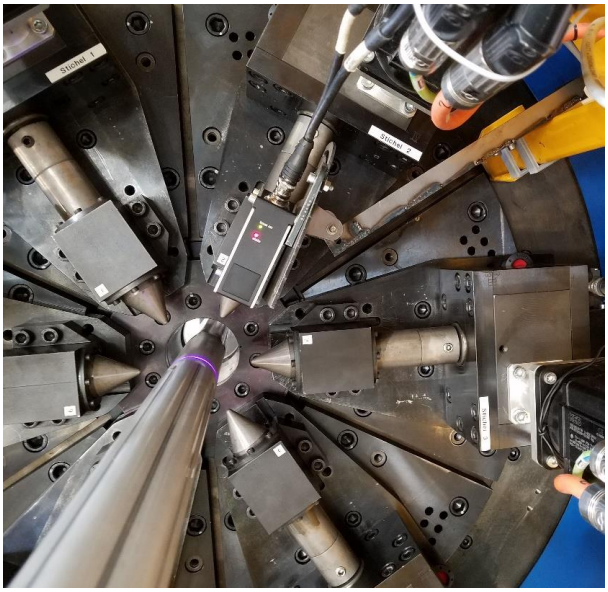

Figure 30: Sensor mount installed on the IPF machine

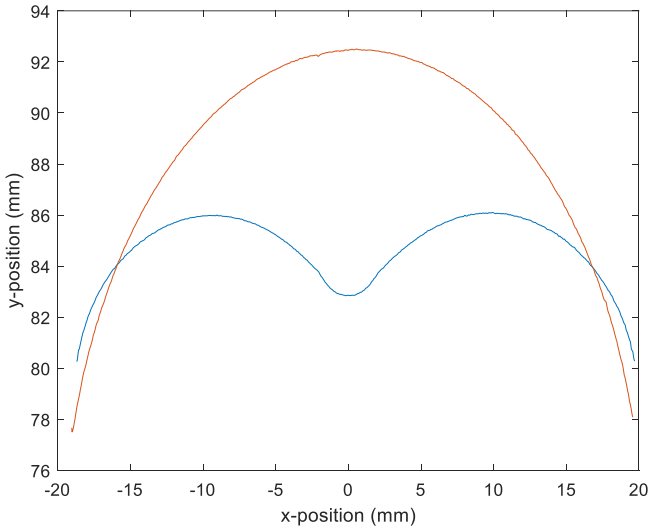

Figure 31: Raw data obtained from the laser triangulation sensor on the custom mount

3.4 Process to measure longitudinal contour geometry

While the optical sensor makes cross-sectional geometry readily, the study of the process mechanics requires that the longitudinal groove geometry be also available while developing models of the process mechanics, though they are not needed for online control. Once the cross-sectional contour at one location along the tube axis is measured using the sensor mount,

methods are also needed to generate longitudinal groove geometry so that this geometry may be compared with the desired groove geometry.

In order to do this, LabVIEW was utilized to write sensor data into a format that can be imported into MATLAB. The data was collected in MATLAB as cross-sectional profile, at a specified location along the tube axis. For each such cross-sectional profile, the lowest radial position or groove depth in the middle of the contour was determined. These low points were then gathered and plotted as the longitudinal groove contour. The flow chart of this procedure is shown in figure 32.

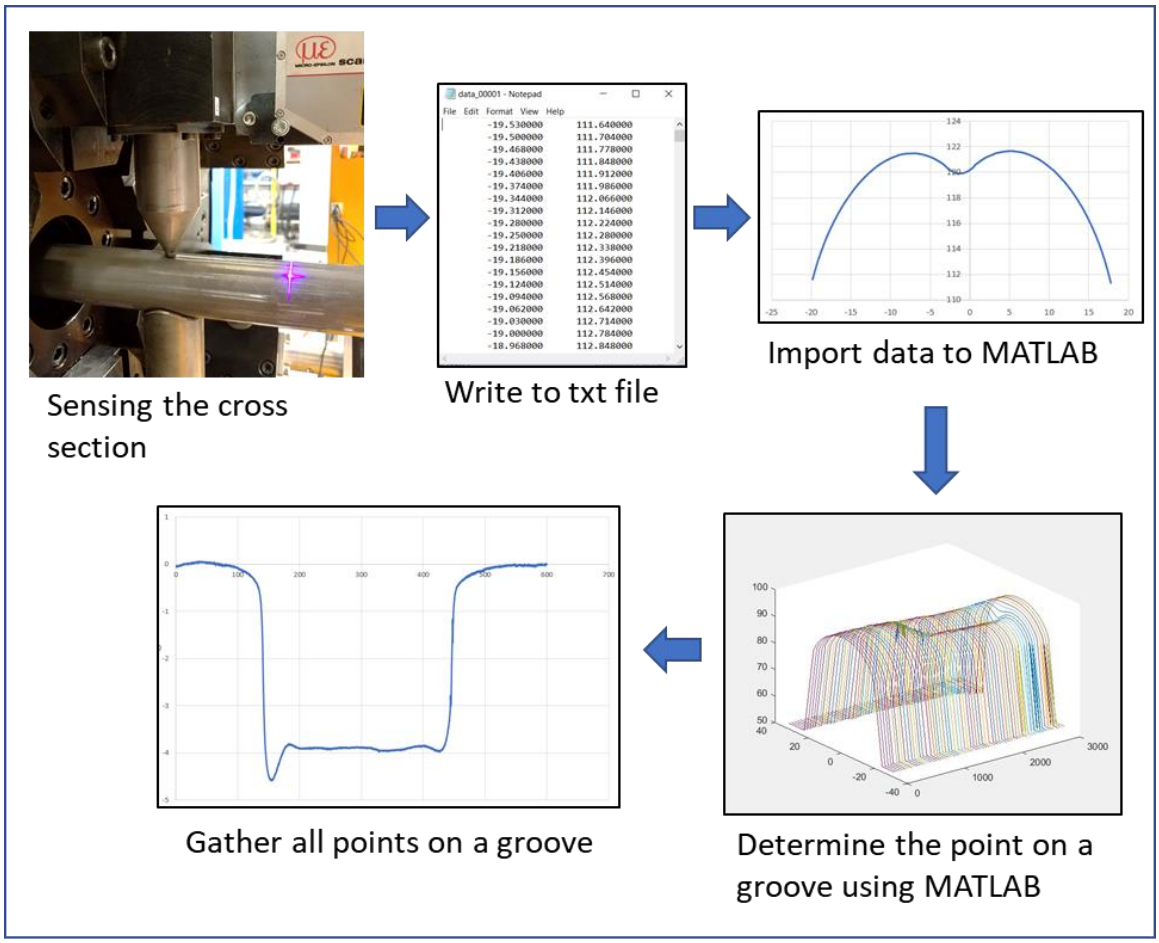

Figure 32: Procedure for generating the longitudinal groove geometry

This procedure offers the benefits of providing both cross-sectional and longitudinal groove geometry data by scanning the groove geometry along the tube only once using the sensor. This increases the variability in what kind of data to obtain from the raw data due to how there will be more than 51200 points per second of data being obtained when this laser triangulation sensor runs at the sampling frequency of 40 Hz. Even though this method works well in capturing both cross-sectional geometry and longitudinal groove geometry, there were problems this methodology faced.

One such problems was that there was a bending of tube that occurred due to the large amount of force that was applied to the tube by the graver tools and apparently a bending moment causing the bending shown in figure 33. This bending caused a measurement problem

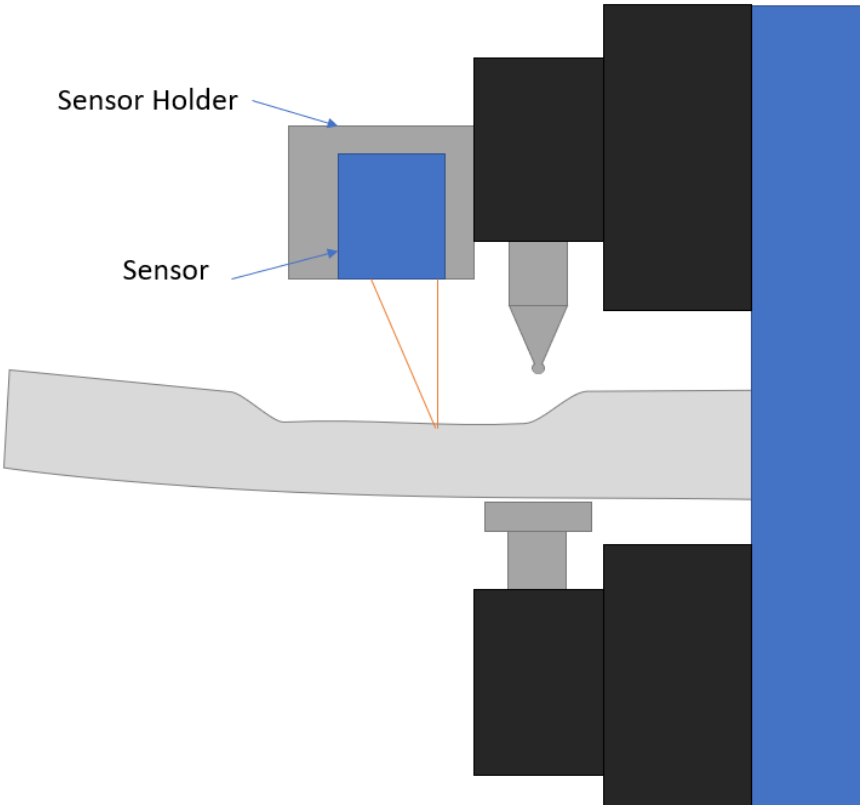

Figure 33: Diagram of tube bending due to forming forces

since it prevented the sensor from measuring groove geometry accurately. Groove geometry during the grooving is measured with reference to the tube geometry measured prior to the forming operation. The underlying problem is either due to misalignment of the un-deformed tube prior to the application of the forming force and the reaction force at the tube support, or due to asymmetric distribution of the reaction force at the tube support, for the case of a single groove represented in figure 33. In either case, the issue calls for appropriate changes in the tube support and holding mechanisms, that is, a change in the structures involved. An alternative solution that could address this issue is to use multiple actively controlled supporting tools to avoid tube bending during the forming operation, based upon sensing of the entire cross-section of the tube during the bending operation. Such a solution would require multiple laser triangulation sensors that measure the whole contour, and some aspects of this measurement will be considered in Chapter 4.

For the current measurement work, there is only a single laser triangulation sensor available, and a software solution to measure longitudinal groove geometry despite tube bending was implemented and is described below. In order to mitigate the effects of tube bending on the measurements, measurements along the undeformed sections of the tube were fit by a circle. Circle fitting was chosen under the assumption that a constant bending moment was acting on these sections of the tube, resulting in a uniform radius of the tube. Thus, circle fitting was done for data processing, by obtaining three points from the contour where the tube is not deformed and fitting a circle through these three points. This was done through use of simple algebra where the center point and radius were found from the three points. By solving the matrix below for x_0 and y_0, the center point can be determined.

$$\begin{bmatrix} 2(x_q - x_p) & 2(y_q - y_p) \\ 2(x_s - x_q) & 2(y_s - y_q) \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} (x_q^2 + y_q^2) - (x_p^2 + y_p^2) \\ (x_s^2 + y_s^2) - (x_q^2 + y_q^2) \end{bmatrix} \quad (1)$$

The center of the circle is $O(x_0, y_0)$and $P(x_p, y_p), Q(x_q, y_q), S(x_s, y_s)$ are the three points selected. By squaring the distances between the center of the circle and any of the three points, the radius of the circle is obtained. Using the circle obtained from circle fitting, the value of the circle fit was subtracted from the original raw data. Figures 34 and 35 show examples of the circle fits that were done for the raw data as well as the processed data after eliminating the effects of the tube bending. The results in the figures confirm the assumed nature of the tube bending since the adjusted tube longitudinal geometry was flat in the undeformed regions, and the depth of the contour could be determined relative to this 'original' tube geometry.

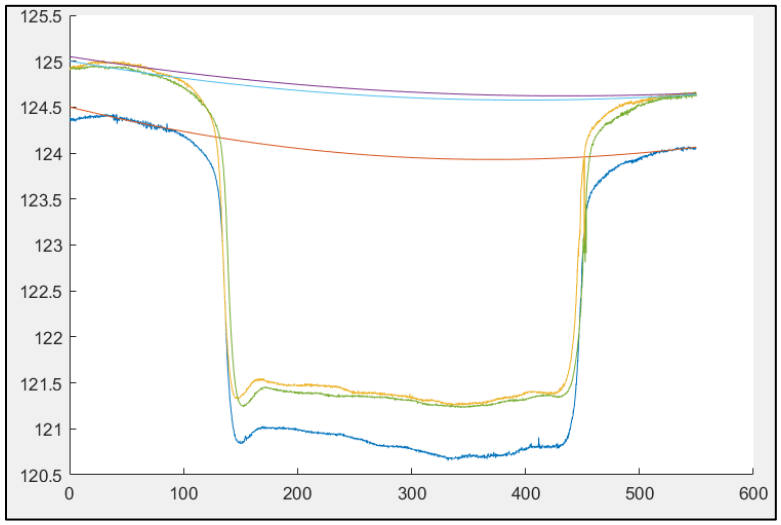

Figure 34: Sample raw data with corresponding circle fits

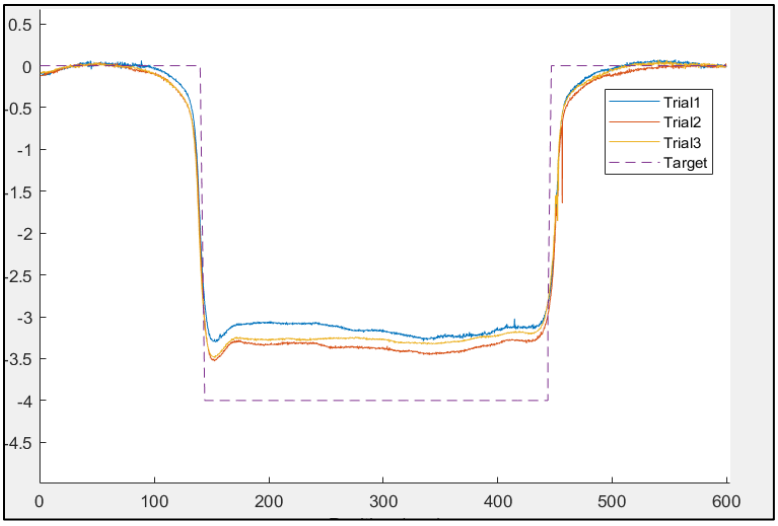

Figure 35: Sample data after the circle fits were applied

3.5 Repeatability of sensor measurements

In order to measure the repeatability of the sensor measurements, different grooves were created and scanned twice with the laser triangulation sensor to determine the repeatability of the measurement alone and that of the process and measurement. The tube used for this experiment is E235 steel with 40 mm diameter and 1mm thickness. For this testing, three 300 mm long axial grooves with 4 mm depth were created on the same tube, at a grooving speed of 10 mm/sec. The programmed tool path is shown in figure 36, and is assumed to be the same as the actual tool path since the tool positioning servos are expected to be very accurate at the low grooving speeds used. The tool used for this experiment was the rigid roller tool, where the tool radius R shown in figure 37 was 4 mm and the D value shown in the figure was 19.5mm. This tool was used for all of the three grooves created in this experiment. Three grooves were formed and measured after the grooves had been created and the tool load removed. The measured grooves are shown in figure 38, where all of the trials were for grooves with the target depth of 4mm.

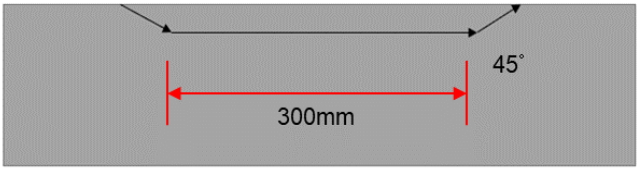

Figure 36: Enlarged contour profile for repeatability experiment

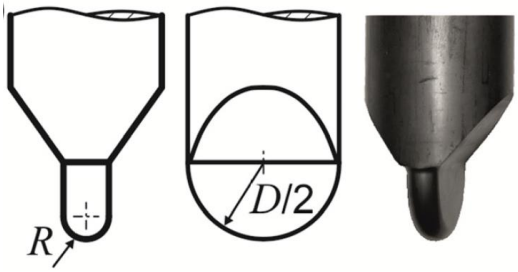

Figure 37: Geometry of the groove used for the repeatability experiment [3]

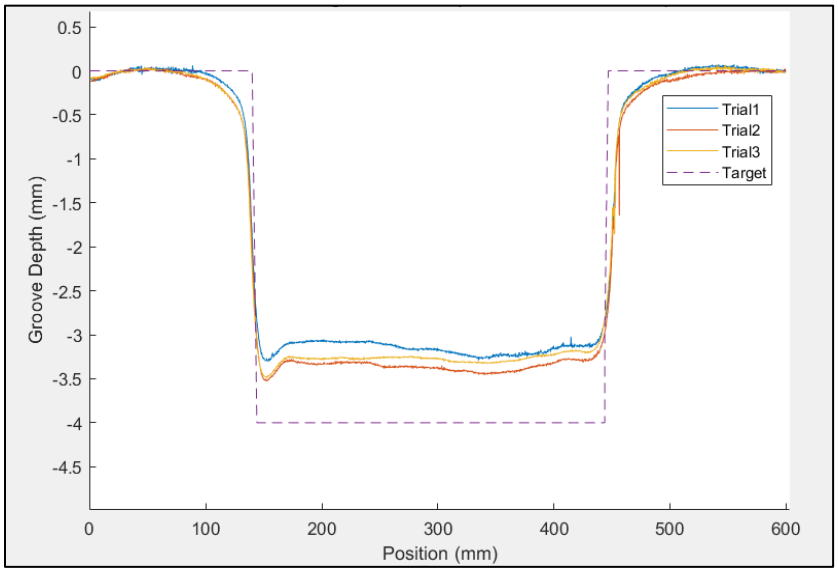

Figure 38: Three different grooves measured to test repeatability of the grooving process

As can be seen in figure 38, the three different trials have different groove depths even though the grooves were all targeted to be 4mm deep, and the tool used was the same with the same tube condition. The variability in the results could be due to lack of repeatability of the process or IPF machine. Also, the grooves were located at different locations along the groove, and may well have resulted in the grooving forces being somewhat different since the boundary

conditions on the tube were not identical. The results in figure 38 therefore represent the combined repeatability of the process and measurement methods.

The three contours were therefore scanned again with the laser triangulation sensor to determine the repeatability of the sensor alone. Figure 39 shows the difference in the two scans for all three grooves. As can be seen the figure, the differences in the two scans are all within $\pm$ 0.1mm. This means that the sensor and the data processing is repeatable and can be used for measuring cross-sectional and longitudinal contours for grooves created by the IPF process.

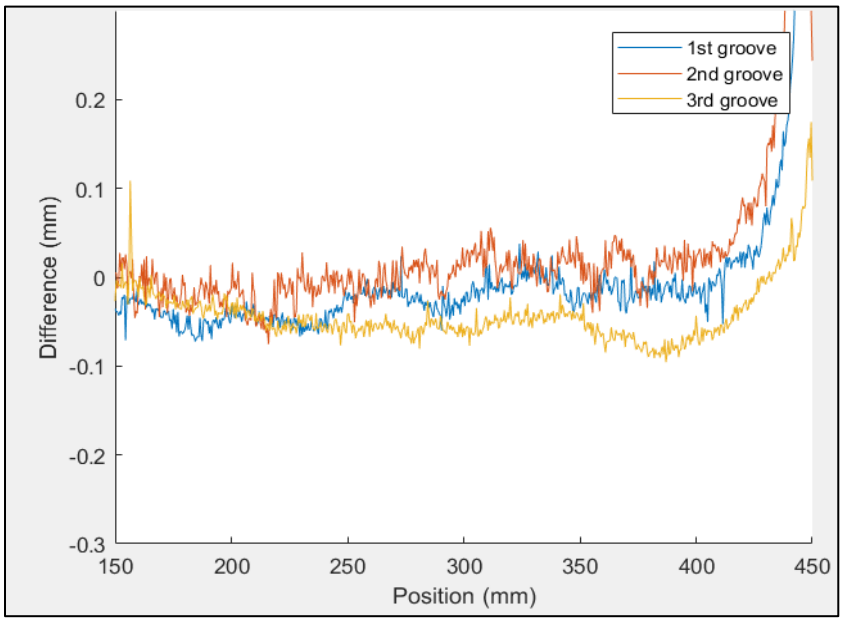

Figure 39: Difference in two measured scans for three different grooves

3.6 Limitations of the laser triangulation sensor

Even though this method seems to work well for the IPF process, there are some limitations that should be noted. One limitation is that the distance of the contour is dependent on both the sensor frequency and the velocity of the tube. Velocity of the tube and sensor frequency can be both set by the user, however, because the longitudinal distance is velocity divide by the laser frequency, the irregularities and some change in the both sensor frequency and the tube velocity will cause an inaccurate determination of the distance in the direction where the tube is

extruding. This needs to be addressed through communication of the sensor and the IPF machine in the future so that the sensor data can be used along with the direction where the tube is extruding.

Another limitation in sensing the longitudinal and cross-sectional geometry is that there is a physical constraint in mounting the sensor on the IPF machine. For the mount created for stable sensor readings, there is a 75 mm gap between the sensor measurement point and the indenter/graver location. A diagram of this is shown in figure 40. As can be seen, this distance is set by the sensor holder. The greater this distance is, the greater is the delay between the indenter action taken and the sensing of the resulting process response. In general, a time delay within the control loop, the less effective is the process control. In addition to this delay, the additional complication for the IPF process, typical of metal forming processes, is that the loaded groove geometry sensed is the geometry under partial loading at the sensor location so that some recovery of elastic deformation would already have taken place. Knowledge of the process mechanics would allow one to address the implications of these sensor limitations for process control more effectively, a topic that is touched upon in Chapter 4 briefly.

Lastly, because the laser triangulation sensor reads 51200 points of data per second when the sensor sampling frequency is 40Hz, there can be some limitations in the speed of writing the files onto the computer. This is because the process of sending the data into MATLAB through the use of LabVIEW requires a transition state of a text file on the computer, and the computer needs to read and write data onto its storage. This can be problematic in the future when multiple sensors are being used to determine the entire tube cross-sectional geometry.

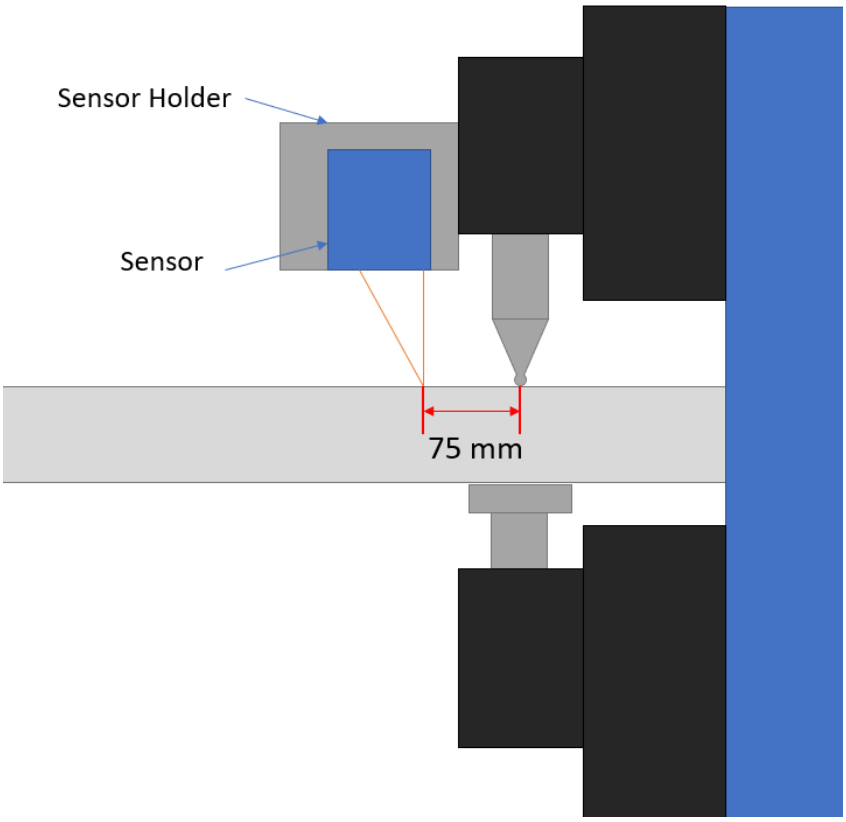

Figure 40: Distance limitation on the mounting of the sensor

3.7: Simulated use of multiple sensors

In order to gather the whole cross-sectional geometry of the tube, multiple laser triangulation sensors would be needed. Since multiple laser triangulation sensors were not available at the TU Dortmund at the time of this work, their use was evaluated by simulation. In order to simulate the use of multiple sensors, different angles of the sensor mounting were used with the single sensor, where the different angles correspond to different sensor locations on the IPF machine with multiple sensors installed. Two different sensor configurations were simulated, where the first simulated 3 sensors mounted 120° apart. The second configuration simulated 6 sensors mounted 60° apart. The two different configurations are shown below in figures 41 and 42.

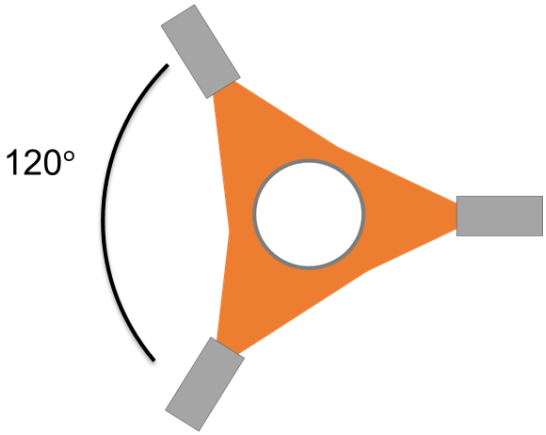

Figure 41: 3-Sensor configuration

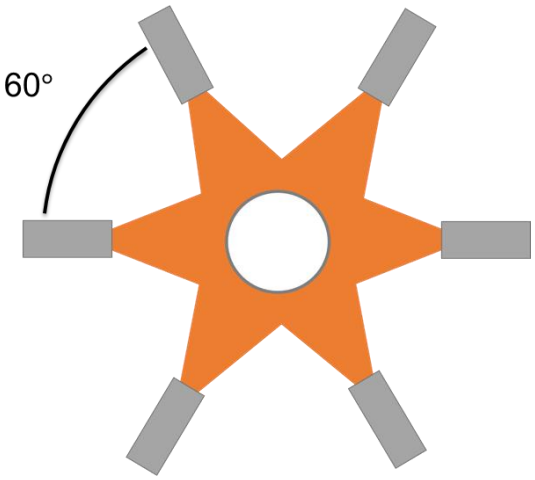

Figure 42: 6-Sensor configuration

In order to test the feasibility of using multiple sensors, a cross-section with 6 grooves was chosen. The specific tube used was an aluminum tube, 44 mm in diameter and 0.6 mm thick. The groove length formed was 500 mm and the sensing was done at the mid-point of the 500 mm long groove. Figures 43 and 44 below show how the 6 gravers created the 6 grooves in the tube.

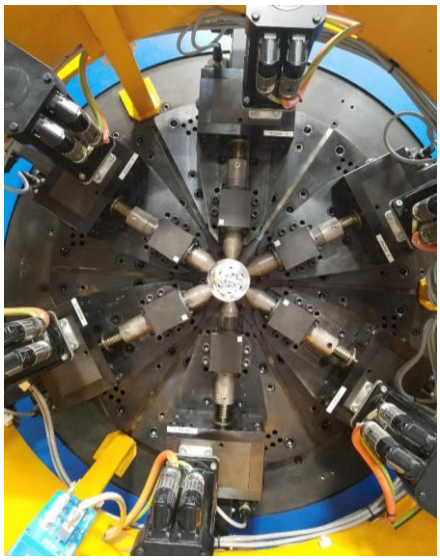

Figure 43: Picture of the 6 installed gravers

Figure 44: Picture of grooves being formed

The sensor was installed after the grooves were formed since the single sensor cannot obtain the geometry of the whole cross-sectional contour during the forming. After the sensor was placed in the IPF machine, the rotary motion of the IPF machine was used to rotate the

sensor into the desired positions. Figure 45 below shows the sensor placement in the IPF machine.

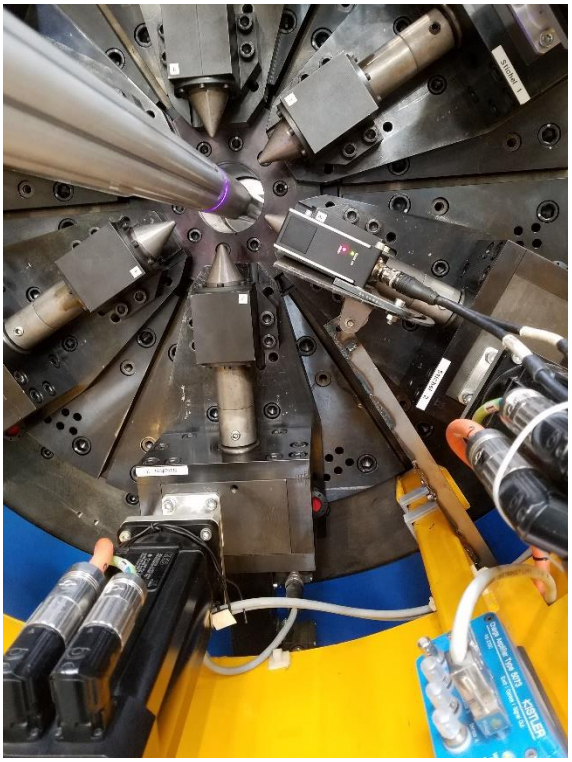

Figure 45: Laser triangulation sensor placed installed on the IPF machine

When the multiple scans were completed, the scans were analyzed in MATLAB. The data was processed with rotational adjustments on the contour, and overlapping contours where different scans captured the same point were identified and matched to create the whole cross-sectional contour. Cross-sectional contours obtained with 3 scans and 6 scans are shown below in figures 46 and 47.

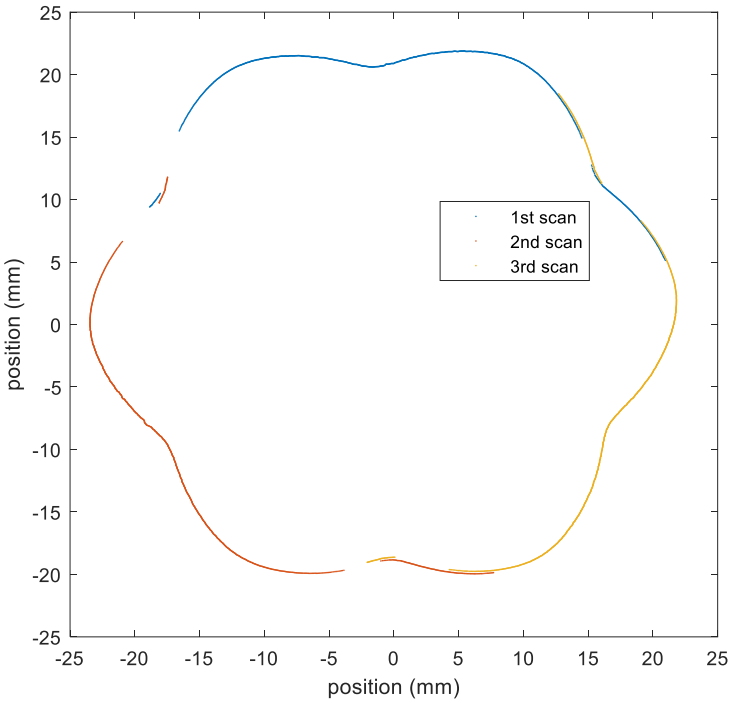

Figure 46: 3 scans with 120° separation

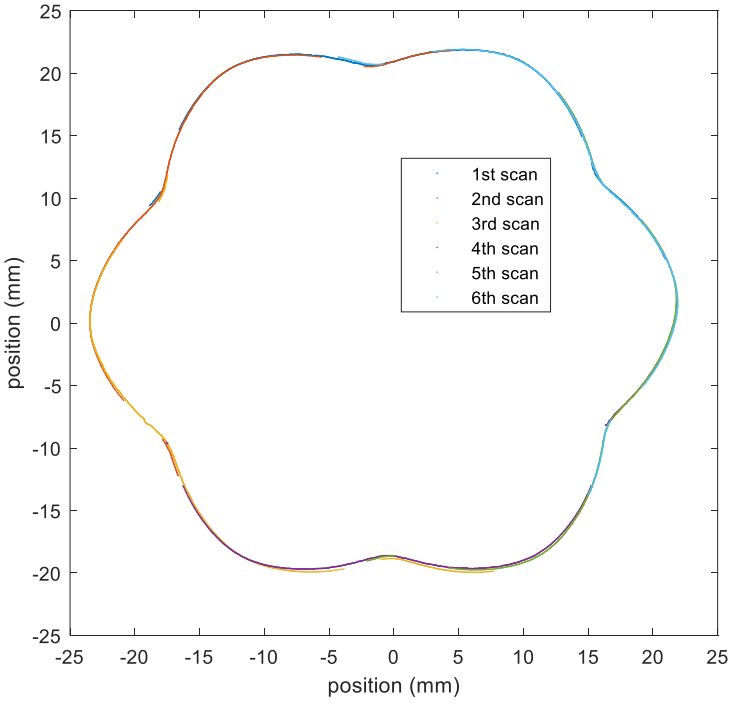

Figure 47: 6 scans with 60° separation

As can be seen in figure 47, the whole contour can be obtained with 6 scans and 60° separation. With 6 scans, the deepest part of the grooves were captured since the sensor was placed on top of the deepest part of the groove. However, with 3 scans and 120° separation, the top left part of the cross-sectional profile is missing due to a shadowing effect. The shadowing effect is caused when the angle of the area on the object being scanned is greater than the angle the sensor can project at. An example of the shadowing effect is shown below in figure 48.

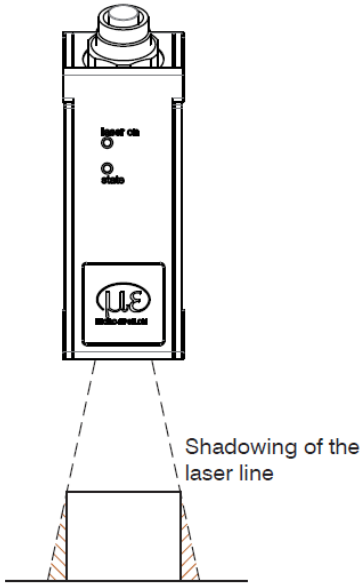

Figure 48: Example of shadowing of the laser line [12]

The shadowing effect caused part of the cross-sectional profile to be missed by the 3-sensor scheme. This is a problem since the whole cross-sectional geometry will need to be sensed and fed back for the process control loop to be effective in controlling geometry. Thus, sensing the whole cross-sectional contour will not be possible with only 3 sensors, and there is a need for more than 3 sensors. As can be seen in figure 47 above, 6 sensors will certainly fulfill the requirement for gathering the whole cross-sectional contour as required for effective process control of geometry. It is true that a smaller number of sensors might also be adequate. However,

given the symmetric nature of the mounting of the six gravers on the IPF machine, six sensors would be easier to incorporate into the machine design than four or five.

Chapter 4: IPF Process Insights from Sensing

4.1 Significance of tool geometry

Online and offline sensing of tube cross-sectional geometry allows one to obtain insights into the IPF process. The effect of tool geometry on cross-sectional and longitudinal groove geometry was investigated. In addition, differences between the partially loaded contours sensed online and the unloaded contours sensed offline after the grooving was completed were investigated as well and insights into the IPF process identified.

Three different tools were used for this experiment. The tools used had hemispherical, rigid roller, and pyramidal geometries, see figure 97. The tool dimensions are shown in figure 47, where the $R_1 = 4\ \text{mm}, D = 39\ \text{mm}, \text{and}\ R_2 = 4\ \text{mm}$. The tool path and contour grooved for this experiment were the same as those used earlier and shown on figure 36, where a 300 mm long, 4mm deep groove was created with tool path entry and exit angles of being $45°$. The tube used for this experiment was an E235 steel tube with 40 mm diameter and 1mm thickness. The support used in all the tests was a 40 mm diameter crescent support.

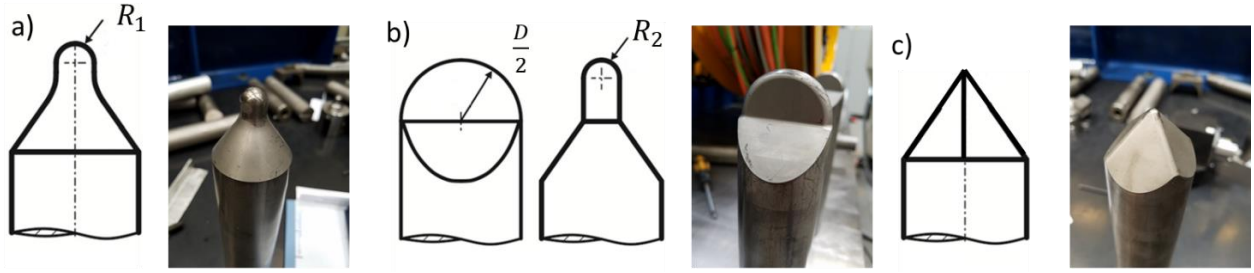

Figure 49: Tool geometries used a) hemispherical b) rigid roller c) pyramidal [3]

4.2 Effect of tool geometry on cross-sectional contour

Cross-sectional geometry was measured for different tool geometries and was compared through analysis done in MATLAB. Figure 50 shows the measured contours for the undeformed tube, prior to grooving, as well as the unloaded cross-sectional contours measured offline after

grooving. The grooved contours were sensed at a location 170 mm away from the start of the groove, which is near the middle of the groove. The profiles of the undeformed tubes were sensed before the grooving in all three cases, and used as references to align the grooved profiles. They are so labeled on the figure. The grooved profiles could then be aligned and compared for the different tool geometries.

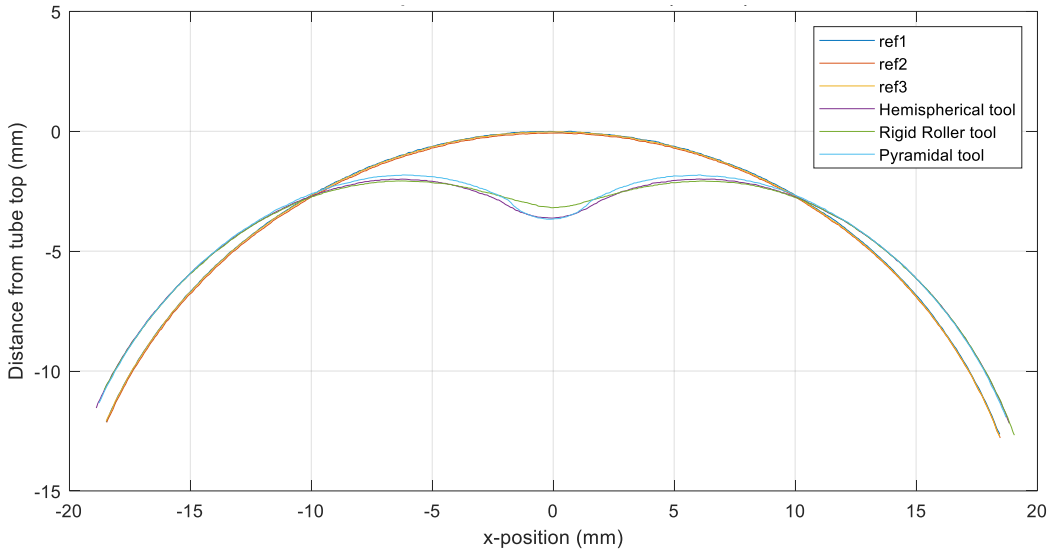

Figure 50: Cross-sectional contours grooved by the different tool geometries

More detailed computational and experimental studies of differences in stress and strain distribution corresponding to different tool geometries have been performed by Grzancic [3].The work described here differs in that online measurements of groove geometry, involving partial loading of the groove, are included for the first time. Further, the tool path trajectory used here is different from that used by Grzancic [3], who used trajectories involving radial indentation followed by axial grooving and radial tool withdrawal.

As may be seen in figure 50, the references for the three tools match the original circular contour well, indicating that all three cases started with practically the same undeformed tube geometry. Thus, the error created by any differences in geometry of the original tube is negligible when compared with the differences in the geometries grooved by the different tools.

Comparing just the groove depths, the hemispherical and pyramidal tools both resulted in about the same depth of 3.65 mm, whereas the rigid roller tool resulted in a smaller groove depth of 3.18 mm. Since the indenter motion during grooving corresponded to a 4 mm depth in all cases, the springback due to recovery of the elastic deformation seems to be larger for the rigid roller case. Examining the groove contours more closely as indicated in figure 51, other differences in their geometries can be seen. For example, the pyramidal tool creates a sharper groove at the location of its deepest point when compared with both the rigid roller tool and the hemispherical tool.

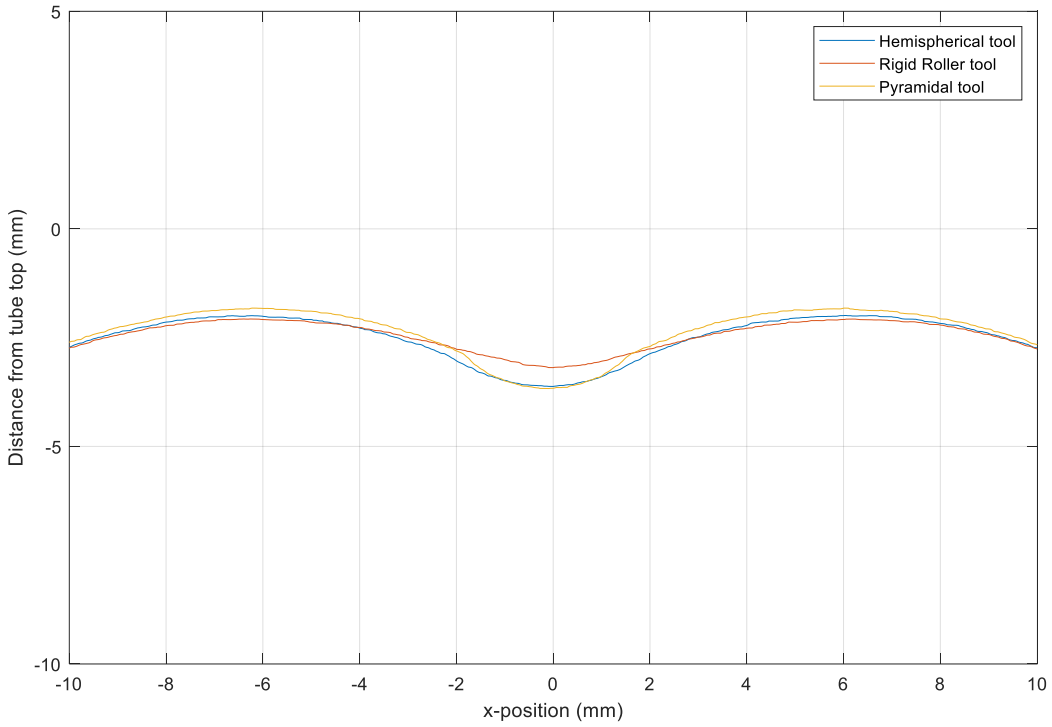

Figure 51: Enlarged cross-sectional contours of the unloaded grooves

The observed differences in groove geometry may be related to differences in the area of contact between the tube and the different tools, which causes differences in the stress fields in both the cross-sectional and longitudinal directions. The experiments indicate that differences in

tool geometry can have a significant impact on groove cross-sectional geometry, and hence tool geometry needs to be considered explicitly in determining the desired cross-sectional geometry.

As shown below in figures 52 and 53, the partially loaded and unloaded cross-sectional contours for the hemispherical and rigid-roller tools were compared with the tool geometry to determine how the tool geometry matches the cross-sectional contours where the tools contact the tube. Tool on the figures are located at a location of -4mm from the top of the tube. Recalling that the partially loaded cross-sectional contour is the contour sensed by the optical sensor when the grooving is in progress, it follows that since the sensed contour is 75 mm away from the indenting plane the loading at the sensed contour is different and less than the loading at the indenting plane. Hence, the corresponding contour is referred to as the partially loaded cross-sectional contour. The unloaded contour is obtained after the grooving is complete.

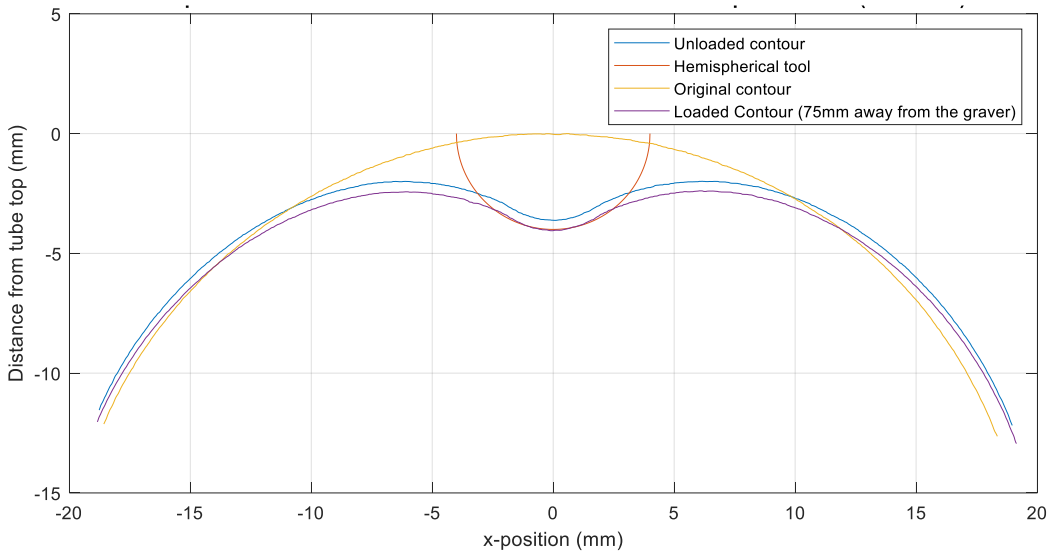

Figure 52: Cross-sectional contours and tool geometry – hemispherical tool

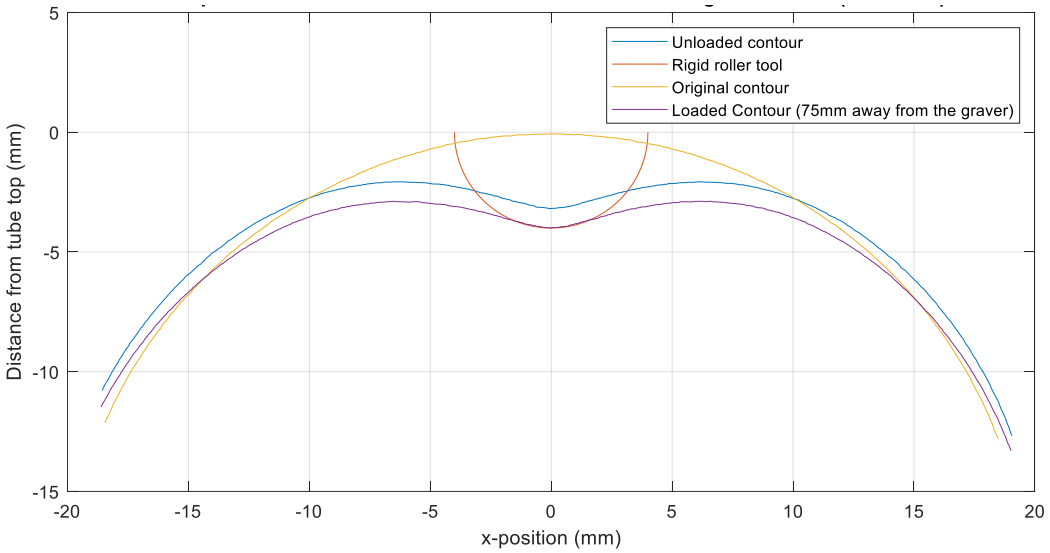

Figure 53: Cross-sectional contours and tool geometry – rigid-roller tool

Figures 52 and 53 indicate that, for both tools, the partially loaded contours are almost the same as the tool contours, suggesting that the elastic recovery at the sensed location is very small, at a distance of 75 mm. The unloaded contours, however, differ significantly from the partially loaded contours, the springback being smaller for the hemispherical tool. Any model for springback in the IPF process would therefore need to provide a rationale for this observation based on process mechanics.

4.3 Differences between partially loaded and unloaded longitudinal contours

Difference between partially loaded and unloaded contours at the deepest points of formed axial grooves was measured for the same three tool geometries mentioned above. The laser triangulation sensor was placed 75 mm away from the graver, thus the contours labeled as "loaded" refer to the sensed contour 75 mm away from where the grooving is occurring. More precisely, the contour will only be partially loaded at the sensed location and not fully loaded. Shown in figures 54 – 56 are partially loaded and unloaded contours sensed by the laser triangulation sensor for the different tool geometries.

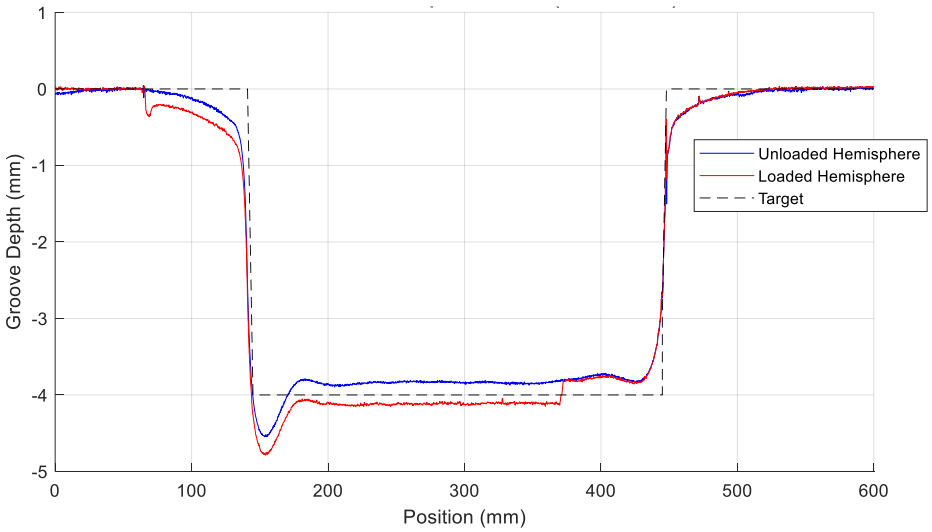

Figure 54: Partially loaded and unloaded longitudinal contours – hemispherical tool

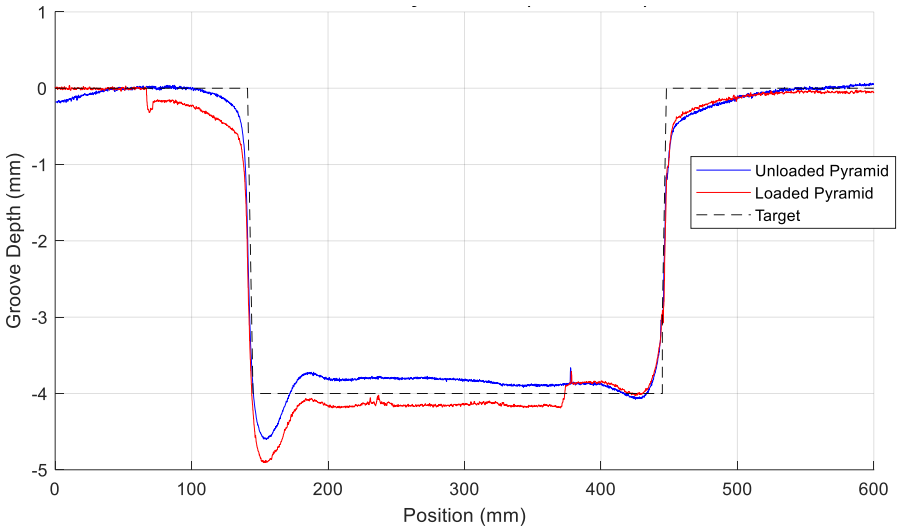

Figure 55: Partially loaded and unloaded longitudinal contours – pyramidal tool

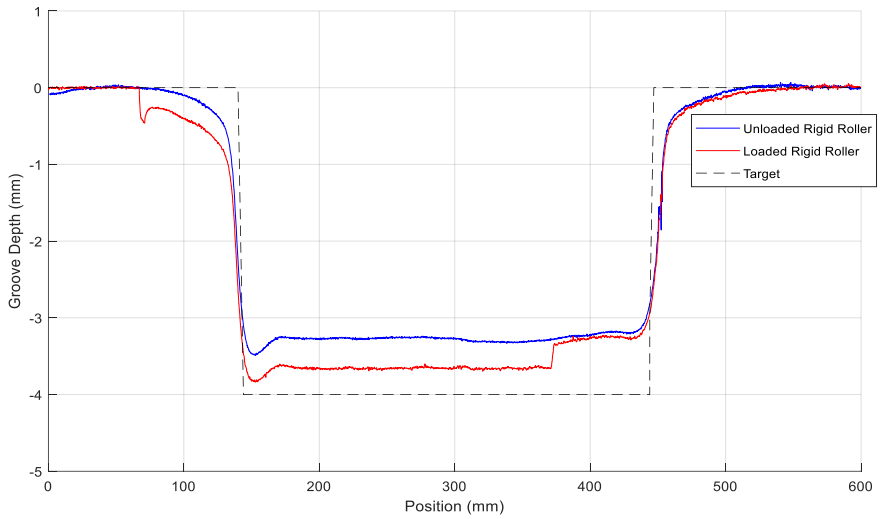

Figure 56: Partially loaded and unloaded longitudinal contours – rigid roller tool

The near-discontinuity in the loaded contours at longitudinal positions of 75 mm and 375 mm occurs when the indenting tool is initiating either the radial indentation or the radial withdrawal, and is most likely due to plastic deformation at the sensing location. The lack of certainty in making this statement arises from the fact that a similar result could also be the result of the tube bending at the time of tool entry or tool exit. If sensors were available to monitor the tube contour at locations diametrically opposite to the groove, the resulting observations would either confirm or disprove occurrence of tube bending. Since a second sensor was not available, a conclusion could not be reached about tube bending. However, since tube bending at the sensor location in other instances were visually observed to be toward the sensor, it seems most likely that the differences in loaded and unloaded contours in figures 54 – 56 were not due to tube bending, but due to plastic deformation at the sensed location.

Figures 54 and 55 indicate that the partially loaded and unloaded longitudinal contours for the hemispherical tool seem to indicate smaller groove depth variations as compared to the pyramidal tool. The latter seems to have a larger overshoot at 180 mm where the unloaded contour rises up from - 4.5 mm to - 3.75 mm. Also, the unloaded contour around 420 mm, when

the tool is being withdrawn, seems to have smaller depth change for the hemispherical tool. Compared to the contours for the hemispherical and pyramidal tools, the rigid roller tool seems to result in smoother depth contours. This could be related to differences in the area of contact of the tool with the tube during forming, as it is the largest for the rigid roller tool. Also, as can be seen in figure 55, the unloaded contour for the rigid roller tool is on average 0.75 mm shallower than the target depth of 4 mm. In order to determine the level of consistency in the amount of difference between the partially loaded and unloaded contour, differences between the partially loaded contours and the unloaded contours are plotted in figure 57 below. As noted earlier, the difference between the partially loaded and unloaded groove was largest for the rigid-roller tool. Hemispherical and pyramidal tools show a similar level of difference, with the hemispherical tool displaying a slightly smaller difference overall. Considering all three cases, there seems to be a trend with the difference decreasing as groove length increases.

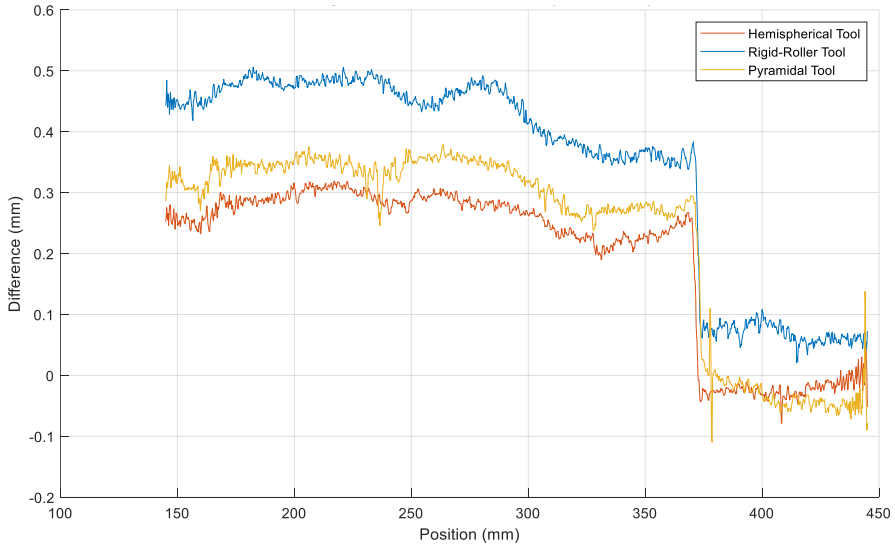

Figure 57: Differences between unloaded and partially loaded groove depth contours

Grzancic [3] has shown results of numerical simulation of the IPF process that are able to predict experimental observations of groove depth contours for a single tool geometry and tool trajectory. He has also produced similar results on cross-sectional contours. The differences in

the cross-sectional and groove depth contours observed here for different tool geometries and for tool trajectories different from that employed by Grzancic suggest a need for more research into, and more understanding of, the broader underlying process mechanics.

Chapter 5: Conclusions and Recommendations for Future Work

5.1 Summary and conclusions

In its current form, the IPF machine has only machine control loops where the indenting tool displacements are controlled based upon continuous feedback of these positions. Control of machine motions does not, however, guarantee control of geometry of the formed part since it is subject to process disturbances such as springback or recovery of elastic deformation. In order to improve control of the part geometry, an additional control loop based upon continuous feedback of part geometry is considered.

In this research, the laser triangulation sensor was selected for the process control loop through literature review and through experimentation, and its suitability for sensing of the profile in IPF processes investigated. As described in chapter 3, the laser triangulation sensor chosen and used here, the Micro-Epsilon scanCONTROL Model 2910-50/BL, had an accuracy of ± 0.1 mm, a repeatability of measurement and data analysis of ± 0.08 mm, and data could be acquired at 40Hz. These characteristics are sufficient for use in the process control loop, if a single groove is to be monitored. If the entire cross-sectional geometry is to be monitored, as would be the case if the profile involved multiple grooves around the tube cross-section, more than three sensors would be needed. Given the geometry of the IPF machine and its six indenters, it is easiest to accommodate six optical sensors in terms of mounting. Evaluation by simulation confirmed that six sensors would be adequate. The sensor was used to examine groove cross-sectional and longitudinal contour measurements during and after grooving with multiple tool geometries, and provided some insights into the mechanics of the IPF process.

The results obtained here also highlighted the need for additional research into the mechanics of the IPF process so that phenomena such as the effect of tool geometry and tool

trajectories on the final geometry of the tube may be understood more generally. The optical sensor chosen here, and the data processing used to extract the data from the raw sensor output, is seen to be effective in generating information in support of such research.

Some limitations of the machine in its current form were also identified during the research. The unsupported section of the tube beyond the indenting plane was seen to bend during indenting, resulting in error in the optical sensor output if it was not compensated. Such compensation for the bending was done by a MATLAB program, which determined the radius of curvature of the tube at the indenting plane based on fitting a circle to the raw data, and adjusting the sensor output accordingly. While this method was adequate for the offline processing used here, it cannot be used for continuous online sensing of the groove geometry. Other solutions need to be determined.

5.2 Recommendations for future work

Through the work described here has confirmed the suitability of the laser triangulation sensor for groove geometry feedback, additional work involving optical sensors that would be needed prior to implementing process control on the IPF machine was identified. Two topics, mounting considerations for multiple sensors in order to sense the entire tube cross-section more effectively, and use of these sensors for active control of the indenters, are addressed here.

As noted in chapter 3, the simulated use of multiple sensors showed that they could be effective for measuring the entire cross-sectional contour around the tube if the number of sensors was adequate. This cannot be done without a sensor mount design to mount all of the multiple sensors. Such mounting design would need to accommodate the space restrictions on the IPF machine. Given the current machine geometry and its six identically mounted indenters, it seems that a six-sensor configuration would be easy to accommodate. It is also desired that the

sensed cross-sectional contour be closer to the graving tool than the current distance of 75 mm. The farther the distance, the greater is the delay between the indenting tool action and the detection of the response of groove geometry to such action and the less effective is feedback control of the process. Thus, an objective of the mounting design is also to sense the groove closer to the indenting plane.

One potential use of the multiple sensors and their ability to sense the entire tube contour is that they can be used to detect bending of the tube during indentation, by making measurements before and after indentation. The use of such information would depend upon the control approach used, and this work would therefore need to be done in conjunction with an appropriate control approach.

Finally, the design of the process control loop based upon sensing of the entire tube cross-sectional contour would depend upon the specifics of the sensed groove geometry, for example, the distance between the sensed groove location and the indentation plane. A variety of process control configurations are expected to be considered, for example, single or multiple grooves, formed with or without helical motion of the indenters around the tube surface. Further, some of the indenters are expected to be used for active support of the tube rather than forming of grooves. The variety of control configurations will result in a variety of demands on sensing and the integration of sensory information to better achieve control goals.

Appendix A: MATLAB Code for Determining the Longitudinal Contour

```
clc
clear
close all

%% Gathering the lowest point of the groove and save
%% ------------------------------user input---------------------------------

freq = 40; %hz
vel = 4; %mm/s
n = 1741; %reference file number
degree = 30; %angle of the sensor

LB = -5; %lowerbound
HB = 2; %higherbound

%folder containing the experimental data files (can be same as above)
str4 ="C:\Users\Ryo\Documents\4 Research\Experiment6_Re_diff_tools";
str5 ="Datap1_2";

%saving file name (make sure to not overwrite)
savefile = 'Datah2_1.xlsx';
% savefile = 'ex2_ref3.xlsx';

%Want to save file?
% Q = true;
Q = false;
%-------------------------------------------------------------------------


%% analysis

%initialize
%find number of files
strin = "*.txt";
totstr = [str4,str5,strin];
filenamein = strjoin(totstr,'\');
filelength = numel(dir(convertStringsToChars(filenamein)))-1;

%initialize other fields
angle = degree*pi()/180;
x = 1:filelength;
R1 = zeros(1,length(x));
S1 = zeros(1,length(x));

s = x.*vel./freq; %changing into mm from velocity and sampling frequency

loss = 0;  %number of data loss

for k = 1:length(x)
    str1 = sprintf("data_%05d.txt",k);
    totstr = [str4,str5,str1];
    filename1 = strjoin(totstr,'\');
    M1 = dlmread(filename1);
    zval1 = (M1(:,2)');
    yval1 = (M1(:,1)');
    idx_temp = find(yval1>LB & yval1<HB);
    zmin_temp = min(zval1(idx_temp));
    num_temp = floor(median(find(zval1 == zmin_temp)));
    yset_temp = yval1(num_temp);

    if yset_temp > HB-0.5 || yset_temp < LB+0.5 %if minimum is not around the center
%         temp = find(yval1>yset-0.01 & yval1<yset+0.01); %save temp file for gathering data
        [n,temp] = max(zval1);

        if length(temp) ~=1
            temp1 = floor(median(temp));
        else
```

```
            temp1 = temp;
        end
    else
        temp1 = num_temp;
    end

    if isnan(temp1) == true %if data is not defined keep as loss
        loss = loss+1;
    else
        R1(k-loss) = zval1(temp1)*cos(angle);
        S1(k-loss) = s(k);
    end

end

%remove zeros
R1(R1==0)=[];
S1(S1==0)=[];

%% Plot
figure(2)
plot(S1,R1)
title('Groove contour by analyzing each profile')
ylabel('displacement (mm)')
xlabel('position (mm)')

%% saving function

if Q == true
    SR = [S1;R1]';
    xlswrite(savefile,SR)
end
```

Appendix B: MATLAB Code for Comparing Longitudinal Geometry with Circle Fit Adjustments

```
%% Experiment 6 Repeatability of springback for different tools

clc
clear
close all
%% decide which analysis to run

hemispherical = 1;
rigid_roller  = 1;
pyramid       = 1;


%% load files

if hemispherical == true
Datah1_1 = xlsread("Datah1_1.xlsx");
Datah1_2 = xlsread("Datah1_2.xlsx");
Datah1_3 = xlsread("Datah1_3.xlsx");
Datah2_1 = xlsread("Datah2_1.xlsx");
Datah2_2 = xlsread("Datah2_2.xlsx");
Datah2_3 = xlsread("Datah2_3.xlsx");
Datah3_1 = xlsread("Datah3_1.xlsx");
Datah3_2 = xlsread("Datah3_2.xlsx");
Datah3_3 = xlsread("Datah3_3.xlsx");
end

if rigid_roller == true
Datar1_1 = xlsread("Datar1_1.xlsx");
Datar1_2 = xlsread("Datar1_2.xlsx");
Datar1_3 = xlsread("Datar1_3.xlsx");
Datar2_1 = xlsread("Datar2_1.xlsx");
Datar2_2 = xlsread("Datar2_2.xlsx");
Datar2_3 = xlsread("Datar2_3.xlsx");
Datar3_1 = xlsread("Datar3_1.xlsx");
Datar3_2 = xlsread("Datar3_2.xlsx");
Datar3_3 = xlsread("Datar3_3.xlsx");
end

if pyramid == true
Datap1_1 = xlsread("Datap1_1.xlsx");
Datap1_2 = xlsread("Datap1_2.xlsx");
Datap1_3 = xlsread("Datap1_3.xlsx");
Datap2_1 = xlsread("Datap2_1.xlsx");
Datap2_2 = xlsread("Datap2_2.xlsx");
Datap2_3 = xlsread("Datap2_3.xlsx");
Datap3_1 = xlsread("Datap3_1.xlsx");
Datap3_2 = xlsread("Datap3_2.xlsx");
Datap3_3 = xlsread("Datap3_3.xlsx");
end

n = 1; %figure numbers

%figures
if hemispherical == true
figure(n)
plot(Datah1_1(:,1),Datah1_1(:,2),Datah1_2(:,1),Datah1_2(:,2),...
     Datah1_3(:,1),Datah1_3(:,2),...
     Datah2_1(:,1),Datah2_1(:,2),Datah2_2(:,1),Datah2_2(:,2),...
     Datah2_3(:,1),Datah2_3(:,2),...
     Datah3_1(:,1),Datah3_1(:,2),Datah3_2(:,1),Datah3_2(:,2),...
     Datah3_3(:,1),Datah3_3(:,2))
legend('1-1','1-2','1-3','2-1','2-2','2-3',...
    '3-1','3-2','3-3')
title('hemispherical')
n=n+1;
end
if rigid_roller == true
```

```
figure(n)
plot(Datar1_1(:,1),Datar1_1(:,2),Datar1_2(:,1),Datar1_2(:,2),...
     Datar1_3(:,1),Datar1_3(:,2),...
     Datar2_1(:,1),Datar2_1(:,2),Datar2_2(:,1),Datar2_2(:,2),...
     Datar2_3(:,1),Datar2_3(:,2),...
     Datar3_1(:,1),Datar3_1(:,2),Datar3_2(:,1),Datar3_2(:,2),...
     Datar3_3(:,1),Datar3_3(:,2))
legend('1-1','1-2','1-3','2-1','2-2','2-3',...
    '3-1','3-2','3-3')
title('rigid roller')
n=n+1;
end
if pyramid == true
figure(n)
plot(Datap1_1(:,1),Datap1_1(:,2),Datap1_2(:,1),Datap1_2(:,2),...
     Datap1_3(:,1),Datap1_3(:,2),...
     Datap2_1(:,1),Datap2_1(:,2),Datap2_2(:,1),Datap2_2(:,2),...
     Datap2_3(:,1),Datap2_3(:,2),...
     Datap3_1(:,1),Datap3_1(:,2),Datap3_2(:,1),Datap3_2(:,2),...
     Datap3_3(:,1),Datap3_3(:,2))
legend('1-1','1-2','1-3','2-1','2-2','2-3',...
    '3-1','3-2','3-3')
title('pyramid')
n=n+1;
end

%% Analysis
if hemispherical == true
for Q = 1

Mlength = 600;

idx1  = find(Datah1_1(:,1)>0 & Datah1_1(:,1)<Mlength);
idx2  = find(Datah1_2(:,1)>0 & Datah1_2(:,1)<Mlength);
idx3  = find(Datah1_3(:,1)>0 & Datah1_3(:,1)<Mlength);

idx4  = find(Datah2_1(:,1)>0 & Datah2_1(:,1)<Mlength);
idx5  = find(Datah2_2(:,1)>0 & Datah2_2(:,1)<Mlength);
idx6  = find(Datah2_3(:,1)>0 & Datah2_3(:,1)<Mlength);

idx7  = find(Datah3_1(:,1)>0 & Datah3_1(:,1)<Mlength);
idx8  = find(Datah3_2(:,1)>0 & Datah3_2(:,1)<Mlength);
idx9  = find(Datah3_3(:,1)>0 & Datah3_3(:,1)<Mlength);


Datah1_1a(:,1) = Datah1_1(idx1,1);
Datah1_1a(:,2) = Datah1_1(idx1,2);
Datah1_2a(:,1) = Datah1_2(idx2,1);
Datah1_2a(:,2) = Datah1_2(idx2,2);
Datah1_3a(:,1) = Datah1_3(idx3,1);
Datah1_3a(:,2) = Datah1_3(idx3,2);
Datah2_1a(:,1) = Datah2_1(idx4,1);
Datah2_1a(:,2) = Datah2_1(idx4,2);
Datah2_2a(:,1) = Datah2_2(idx5,1);
Datah2_2a(:,2) = Datah2_2(idx5,2);
Datah2_3a(:,1) = Datah2_3(idx6,1);
Datah2_3a(:,2) = Datah2_3(idx6,2);
Datah3_1a(:,1) = Datah3_1(idx7,1);
Datah3_1a(:,2) = Datah3_1(idx7,2);
Datah3_2a(:,1) = Datah3_2(idx8,1);
Datah3_2a(:,2) = Datah3_2(idx8,2);
Datah3_3a(:,1) = Datah3_3(idx9,1);
Datah3_3a(:,2) = Datah3_3(idx9,2);

%circle fit
s1 = 200;
s2 = 400;
x1(1) = Datah1_3a(s1,1);
x1(2) = Datah1_3a(end-200,1);
x1(3) = Datah1_3a(s2,1);
y1(1) = Datah1_3a(s1,2);
```

```
y1(2) = Datah1_3a(end-200,2);
y1(3) = Datah1_3a(s2,2);

x2(1) = Datah2_3a(s1,1);
x2(2) = Datah2_3a(end-200,1);
x2(3) = Datah2_3a(s2,1);
y2(1) = Datah2_3a(s1,2);
y2(2) = Datah2_3a(end-200,2);
y2(3) = Datah2_3a(s2,2);

x3(1) = Datah3_3a(s1,1);
x3(2) = Datah3_3a(end-200,1);
x3(3) = Datah3_3a(s2,1);
y3(1) = Datah3_3a(s1,2);
y3(2) = Datah3_3a(end-200,2);
y3(3) = Datah3_3a(s2,2);

 [R,xcyc] = fit_circle_through_3_points([x1' y1' x2' y2' x3' y3']);
yvals1 = xcyc(2,1)+sqrt(R(1)^2-(Datah1_3a(:,1)-xcyc(1,1)).^2);
yvals2 = xcyc(2,2)+sqrt(R(2)^2-(Datah2_3a(:,1)-xcyc(1,2)).^2);
yvals3 = xcyc(2,3)+sqrt(R(3)^2-(Datah3_3a(:,1)-xcyc(1,3)).^2);

if abs(yvals1(1)-y1(1))>100
    yvals1 =  xcyc(2,1)-sqrt(R(1)^2-(Datah1_3a(:,1)-xcyc(1,1)).^2);
end
if abs(yvals2(1)-y2(1))>100
    yvals2 =  xcyc(2,2)-sqrt(R(2)^2-(Datah2_3a(:,1)-xcyc(1,2)).^2);
end
if abs(yvals3(1)-y3(1))>100
    yvals3 =  xcyc(2,3)-sqrt(R(3)^2-(Datah3_3a(:,1)-xcyc(1,3)).^2);
end


figure(n)
hold on
plot(Datah1_3a(:,1),Datah1_3a(:,2),Datah1_3a(:,1),yvals1)
plot(Datah2_3a(:,1),Datah2_3a(:,2),Datah2_3a(:,1),yvals2)
plot(Datah3_3a(:,1),Datah3_3a(:,2),Datah3_3a(:,1),yvals3)
hold off
n=n+1;


%difference of circle fit and original
cdiff1 = Datah1_3a(:,2) - yvals1;
cdiff2 = Datah2_3a(:,2) - yvals2;
cdiff3 = Datah3_3a(:,2) - yvals3;

%drawing target line
refx1 = 0:140;
refx2 = 141:145;
refx3 = 145:445;
refx4 = 445:448;
refx5 = 449:600;

refy1 = 0*ones(1,length(refx1));
refy2 = -refx2+refx2(1);
refy3 = -4*ones(1,length(refx3));
refy4 = refx4-refx4(end);
refy5 = 0*ones(1,length(refx5));

refx = [refx1 refx2 refx3 refx4 refx5];
refy = [refy1 refy2 refy3 refy4 refy5];

figure(n)
hold on
plot(Datah1_3a(:,1),cdiff1)
plot(Datah2_3a(:,1),cdiff2)
plot(Datah3_3a(:,1),cdiff3)
plot(refx,refy,'--')
legend('Trial1','Trial2','Trial3','Target')
title('Contours of Hemispherical tool (Unloaded, 4mm indent)')
```

```
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n = n+1;

%difference of loaded and original

dx = 0:0.1:Mlength;
dy1 = zeros(1,length(dx));
dy2 = zeros(1,length(dx));
dy3 = zeros(1,length(dx));
for k = 1:length(dx)
    tempidx1 = Datah1_1a(:,1) > dx(k)-0.25 & Datah1_1a(:,1) < dx(k)+0.25;
    tempidx2 = Datah1_2a(:,1) > dx(k)-0.25 & Datah1_2a(:,1) < dx(k)+0.25;

    tempidx3 = Datah2_1a(:,1) > dx(k)-0.25 & Datah2_1a(:,1) < dx(k)+0.25;
    tempidx4 = Datah2_2a(:,1) > dx(k)-0.25 & Datah2_2a(:,1) < dx(k)+0.25;

    tempidx5 = Datah3_1a(:,1) > dx(k)-0.25 & Datah3_1a(:,1) < dx(k)+0.25;
    tempidx6 = Datah3_2a(:,1) > dx(k)-0.25 & Datah3_2a(:,1) < dx(k)+0.25;

    dy1(k) = median(Datah1_2a((tempidx2),2))-median(Datah1_1a((tempidx1),2));
    dy2(k) = median(Datah2_2a((tempidx4),2))-median(Datah2_1a((tempidx3),2));
    dy3(k) = median(Datah3_2a((tempidx6),2))-median(Datah3_1a((tempidx5),2));
end
dx1 = 0:0.1:Mlength;
dx2 = 0:0.1:Mlength;
dx3 = 0:0.1:Mlength;

dx1(find(isnan(dy1)))= [];
dy1(find(isnan(dy1)))= [];
dx2(find(isnan(dy2)))= [];
dy2(find(isnan(dy2)))= [];
dx3(find(isnan(dy3)))= [];
dy3(find(isnan(dy3)))= [];

figure(n)
hold on
plot(dx1,dy1)
plot(dx2,dy2)
plot(dx3,dy3)
plot(refx,refy,'--')
legend('Loaded Hemisphere trial1','Loaded Hemisphere trial2',...
    'Loaded Hemisphere trial3','Target')
title('Contous of Hemispherical tool (Loaded, 4mm indent)')
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n = n+1;
%unforced and forced difference
figure(n)
hold on
plot(Datah1_3a(:,1),cdiff1)
plot(Datah2_3a(:,1),cdiff2)
plot(Datah3_3a(:,1),cdiff3)
plot(dx1,dy1)
plot(dx2,dy2)
plot(dx3,dy3)
plot(refx,refy,'--')
legend('Unloaded Hemisphere trial1','Unloaded Hemisphere trial2',...
    'Unloaded Hemisphere trial3'...
   ,'Loaded Hemisphere trial1','Loaded Hemisphere trial2',...
    'Loaded Hemisphere trial3','Target')
title('Contours of Hemispherical tool (4mm indent)')
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n = n+1;

glength = 300;
st = 145;
```

```
dxa = st:0.2:(st+glength);
dy4 = zeros(1,length(dxa));
dy5 = zeros(1,length(dxa));
dy6 = zeros(1,length(dxa));
dy7 = zeros(1,length(dxa));
dy8 = zeros(1,length(dxa));
dy9 = zeros(1,length(dxa));

dx1 = dx1';
dx2 = dx2';
dx3 = dx3';
dy1 = dy1';
dy2 = dy2';
dy3 = dy3';
for k = 1:length(dxa)
    tempidx7 = dx1 > dxa(k)-0.5 & dx1 < dxa(k)+0.5;
    tempidx8 = Datah1_3a(:,1) > dxa(k)-0.5 & Datah1_3a(:,1) < dxa(k)+0.5;

    tempidx9 = dx2 > dxa(k)-0.5 & dx2 < dxa(k)+0.5;
    tempidx10 = Datah2_3a(:,1) > dxa(k)-0.5 & Datah2_3a(:,1) < dxa(k)+0.5;

    tempidx11 = dx3 > dxa(k)-0.5 & dx3 < dxa(k)+0.5;
    tempidx12 = Datah3_3a(:,1) > dxa(k)-0.5 & Datah3_3a(:,1) < dxa(k)+0.5;

    dy4(k) = (median(cdiff1(tempidx8)) - median(dy1(tempidx7)));
    dy5(k) = (median(cdiff2(tempidx10))- median(dy2(tempidx9)));
    dy6(k) = (median(cdiff3(tempidx12))- median(dy3(tempidx11)));
    dy7(k) = (-4 - median(cdiff1(tempidx8)))/-4*100;
    dy8(k) = (-4 - median(cdiff2(tempidx10)))/-4*100;
    dy9(k) = (-4 - median(cdiff3(tempidx12)))/-4*100;
end

dx4 = st:0.2:(st+glength);
dx5 = st:0.2:(st+glength);
dx6 = st:0.2:(st+glength);

dx7 = st:0.2:(st+glength);
dx8 = st:0.2:(st+glength);
dx9 = st:0.2:(st+glength);

dx4(find(isnan(dy4)))= [];
dy4(find(isnan(dy4)))= [];
dx5(find(isnan(dy5)))= [];
dy5(find(isnan(dy5)))= [];
dx6(find(isnan(dy6)))= [];
dy6(find(isnan(dy6)))= [];

dx7(find(isnan(dy7)))= [];
dy7(find(isnan(dy7)))= [];
dx8(find(isnan(dy8)))= [];
dy8(find(isnan(dy8)))= [];
dx9(find(isnan(dy9)))= [];
dy9(find(isnan(dy9)))= [];

figure(n)
hold on
plot(dx4,dy4)
plot(dx5,dy5)
plot(dx6,dy6)
legend('Trial1','Trial2','Trial3')
title('Springback of Hemispherical tool (4mm indent)')
xlabel('Position (mm)')
ylabel('Difference (mm)')
hold off
n = n+1;


figure(n)
hold on
plot(dx7,dy7)
plot(dx8,dy8)
```

```
plot(dx9,dy9)
legend('Trial1','Trial2','Trial3')
title('Springback of Hemispherical tool (ref 4mm indent)')
xlabel('Position (mm)')
ylabel('Percent Difference (%)')
hold off
n = n+1;
end
end

if rigid_roller == true
for Q = 1

Mlength = 600;

idx1  = find(Datar1_1(:,1)>0 & Datar1_1(:,1)<Mlength);
idx2  = find(Datar1_2(:,1)>0 & Datar1_2(:,1)<Mlength);
idx3  = find(Datar1_3(:,1)>0 & Datar1_3(:,1)<Mlength);

idx4  = find(Datar2_1(:,1)>0 & Datar2_1(:,1)<Mlength);
idx5  = find(Datar2_2(:,1)>0 & Datar2_2(:,1)<Mlength);
idx6  = find(Datar2_3(:,1)>0 & Datar2_3(:,1)<Mlength);

idx7  = find(Datar3_1(:,1)>0 & Datar3_1(:,1)<Mlength);
idx8  = find(Datar3_2(:,1)>0 & Datar3_2(:,1)<Mlength);
idx9  = find(Datar3_3(:,1)>0 & Datar3_3(:,1)<Mlength);


Datar1_1a(:,1) = Datar1_1(idx1,1);
Datar1_1a(:,2) = Datar1_1(idx1,2);
Datar1_2a(:,1) = Datar1_2(idx2,1);
Datar1_2a(:,2) = Datar1_2(idx2,2);
Datar1_3a(:,1) = Datar1_3(idx3,1);
Datar1_3a(:,2) = Datar1_3(idx3,2);
Datar2_1a(:,1) = Datar2_1(idx4,1);
Datar2_1a(:,2) = Datar2_1(idx4,2);
Datar2_2a(:,1) = Datar2_2(idx5,1);
Datar2_2a(:,2) = Datar2_2(idx5,2);
Datar2_3a(:,1) = Datar2_3(idx6,1);
Datar2_3a(:,2) = Datar2_3(idx6,2);
Datar3_1a(:,1) = Datar3_1(idx7,1);
Datar3_1a(:,2) = Datar3_1(idx7,2);
Datar3_2a(:,1) = Datar3_2(idx8,1);
Datar3_2a(:,2) = Datar3_2(idx8,2);
Datar3_3a(:,1) = Datar3_3(idx9,1);
Datar3_3a(:,2) = Datar3_3(idx9,2);

%circle fit
s1 = 200;
s2 = 400;
x1(1) = Datar1_3a(s1,1);
x1(2) = Datar1_3a(end-45,1);
x1(3) = Datar1_3a(s2,1);
y1(1) = Datar1_3a(s1,2);
y1(2) = Datar1_3a(end-45,2);
y1(3) = Datar1_3a(s2,2);

x2(1) = Datar2_3a(s1,1);
x2(2) = Datar2_3a(end-45,1);
x2(3) = Datar2_3a(s2,1);
y2(1) = Datar2_3a(s1,2);
y2(2) = Datar2_3a(end-45,2);
y2(3) = Datar2_3a(s2,2);

x3(1) = Datar3_3a(s1,1);
x3(2) = Datar3_3a(end-45,1);
x3(3) = Datar3_3a(s2,1);
y3(1) = Datar3_3a(s1,2);
y3(2) = Datar3_3a(end-45,2);
y3(3) = Datar3_3a(s2,2);
```

```
 [R,xcyc] = fit_circle_through_3_points([x1' y1' x2' y2' x3' y3']);
yvals1 = xcyc(2,1)+sqrt(R(1)^2-(Datar1_3a(:,1)-xcyc(1,1)).^2);
yvals2 = xcyc(2,2)+sqrt(R(2)^2-(Datar2_3a(:,1)-xcyc(1,2)).^2);
yvals3 = xcyc(2,3)+sqrt(R(3)^2-(Datar3_3a(:,1)-xcyc(1,3)).^2);

if abs(yvals1(1)-y1(1))>100
    yvals1 =  xcyc(2,1)-sqrt(R(1)^2-(Datar1_3a(:,1)-xcyc(1,1)).^2);
end
if abs(yvals2(1)-y2(1))>100
    yvals2 =  xcyc(2,2)-sqrt(R(2)^2-(Datar2_3a(:,1)-xcyc(1,2)).^2);
end
if abs(yvals3(1)-y3(1))>100
    yvals3 =  xcyc(2,3)-sqrt(R(3)^2-(Datar3_3a(:,1)-xcyc(1,3)).^2);
end


figure(n)
hold on
plot(Datar1_3a(:,1),Datar1_3a(:,2),Datar1_3a(:,1),yvals1)
plot(Datar2_3a(:,1),Datar2_3a(:,2),Datar2_3a(:,1),yvals2)
plot(Datar3_3a(:,1),Datar3_3a(:,2),Datar3_3a(:,1),yvals3)
hold off
n=n+1;


%difference of circle fit and original
cdiff1 = Datar1_3a(:,2) - yvals1;
cdiff2 = Datar2_3a(:,2) - yvals2;
cdiff3 = Datar3_3a(:,2) - yvals3;

%drawing target line
refx1 = 0:139;
refx2 = 140:144;
refx3 = 144:444;
refx4 = 444:447;
refx5 = 448:600;

refy1 = 0*ones(1,length(refx1));
refy2 = -refx2+refx2(1);
refy3 = -4*ones(1,length(refx3));
refy4 = refx4-refx4(end);
refy5 = 0*ones(1,length(refx5));

refx = [refx1 refx2 refx3 refx4 refx5];
refy = [refy1 refy2 refy3 refy4 refy5];

figure(n)
hold on
plot(Datar1_3a(:,1),cdiff1)
plot(Datar2_3a(:,1),cdiff2)
plot(Datar3_3a(:,1),cdiff3)
plot(refx,refy,'--')
legend('Trial1','Trial2','Trial3','Target')
title('Contour of Rigid roller tool (Unloaded, 4mm indent)')
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n = n+1;

%difference of loaded and original

dx = 0:0.1:Mlength;
dy1 = zeros(1,length(dx));
dy2 = zeros(1,length(dx));
dy3 = zeros(1,length(dx));
for k = 1:length(dx)
    tempidx1 = Datar1_1a(:,1) > dx(k)-0.25 & Datar1_1a(:,1) < dx(k)+0.25;
    tempidx2 = Datar1_2a(:,1) > dx(k)-0.25 & Datar1_2a(:,1) < dx(k)+0.25;

    tempidx3 = Datar2_1a(:,1) > dx(k)-0.25 & Datar2_1a(:,1) < dx(k)+0.25;
    tempidx4 = Datar2_2a(:,1) > dx(k)-0.25 & Datar2_2a(:,1) < dx(k)+0.25;
```

```
    tempidx5 = Datar3_1a(:,1) > dx(k)-0.25 & Datar3_1a(:,1) < dx(k)+0.25;
    tempidx6 = Datar3_2a(:,1) > dx(k)-0.25 & Datar3_2a(:,1) < dx(k)+0.25;

    dy1(k) = median(Datar1_2a((tempidx2),2))-median(Datar1_1a((tempidx1),2));
    dy2(k) = median(Datar2_2a((tempidx4),2))-median(Datar2_1a((tempidx3),2));
    dy3(k) = median(Datar3_2a((tempidx6),2))-median(Datar3_1a((tempidx5),2));
end
dx1 = 0:0.1:Mlength;
dx2 = 0:0.1:Mlength;
dx3 = 0:0.1:Mlength;

dx1(find(isnan(dy1)))= [];
dy1(find(isnan(dy1)))= [];
dx2(find(isnan(dy2)))= [];
dy2(find(isnan(dy2)))= [];
dx3(find(isnan(dy3)))= [];
dy3(find(isnan(dy3)))= [];

figure(n)
hold on
plot(dx1,dy1)
plot(dx2,dy2)
plot(dx3,dy3)
plot(refx,refy,'--')
legend('Trial1','Trial2','Trial3','Target')
title('Contour of rigid roller tool (Loaded, 4mm indent)')
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n= n+1;

%unforced and forced difference
figure(n)
hold on
plot(Datar1_3a(:,1),cdiff1)
plot(Datar2_3a(:,1),cdiff2)
plot(Datar3_3a(:,1),cdiff3)
plot(dx1,dy1)
plot(dx2,dy2)
plot(dx3,dy3)
plot(refx,refy,'--')
legend('Unloaded Rigid Roller trial1','Unloaded Rigid Roller trial2',...
    'Unloaded Rigid Roller trial3'...
   ,'Loaded Rigid Roller trial1','Loaded Rigid Roller trial2',...
   'Loaded Rigid Roller trial3','Target')
title('Contour of rigid roller tool (4mm indent)')
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n = n+1;

glength = 300;
st = 145;
dxa = st:0.2:(st+glength);
dy4 = zeros(1,length(dxa));
dy5 = zeros(1,length(dxa));
dy6 = zeros(1,length(dxa));
dy7 = zeros(1,length(dxa));
dy8 = zeros(1,length(dxa));
dy9 = zeros(1,length(dxa));

dx1 = dx1';
dx2 = dx2';
dx3 = dx3';
dy1 = dy1';
dy2 = dy2';
dy3 = dy3';
for k = 1:length(dxa)
    tempidx7 = dx1 > dxa(k)-0.5 & dx1 < dxa(k)+0.5;
    tempidx8 = Datar1_3a(:,1) > dxa(k)-0.5 & Datar1_3a(:,1) < dxa(k)+0.5;
```

```
    tempidx9 = dx2 > dxa(k)-0.5 & dx2 < dxa(k)+0.5;
    tempidx10 = Datar2_3a(:,1) > dxa(k)-0.5 & Datar2_3a(:,1) < dxa(k)+0.5;

    tempidx11 = dx3 > dxa(k)-0.5 & dx3 < dxa(k)+0.5;
    tempidx12 = Datar3_3a(:,1) > dxa(k)-0.5 & Datar3_3a(:,1) < dxa(k)+0.5;

    dy4(k) = (median(cdiff1(tempidx8)) - median(dy1(tempidx7)));
    dy5(k) = (median(cdiff2(tempidx10))- median(dy2(tempidx9)));
    dy6(k) = (median(cdiff3(tempidx12))- median(dy3(tempidx11)));
    dy7(k) = (-4 - median(cdiff1(tempidx8)))/-4*100;
    dy8(k) = (-4 - median(cdiff2(tempidx10)))/-4*100;
    dy9(k) = (-4 - median(cdiff3(tempidx12)))/-4*100;
end

dx4 = st:0.2:(st+glength);
dx5 = st:0.2:(st+glength);
dx6 = st:0.2:(st+glength);

dx7 = st:0.2:(st+glength);
dx8 = st:0.2:(st+glength);
dx9 = st:0.2:(st+glength);

dx4(find(isnan(dy4)))= [];
dy4(find(isnan(dy4)))= [];
dx5(find(isnan(dy5)))= [];
dy5(find(isnan(dy5)))= [];
dx6(find(isnan(dy6)))= [];
dy6(find(isnan(dy6)))= [];

dx7(find(isnan(dy7)))= [];
dy7(find(isnan(dy7)))= [];
dx8(find(isnan(dy8)))= [];
dy8(find(isnan(dy8)))= [];
dx9(find(isnan(dy9)))= [];
dy9(find(isnan(dy9)))= [];

figure(n)
hold on
plot(dx4,dy4)
plot(dx5,dy5)
plot(dx6,dy6)
legend('Trial1','Trial2','Trial3')
title('Springback of Rigid Roller tool (4mm indent)')
xlabel('Position (mm)')
ylabel('Difference (mm)')
hold off
n = n+1;

figure(n)
hold on
plot(dx7,dy7)
plot(dx8,dy8)
plot(dx9,dy9)
legend('Trial1','Trial2','Trial3')
title('Springback of Rigid roller tool (ref 4mm indent)')
xlabel('Position (mm)')
ylabel('Percent Difference (%)')
hold off
n = n+1;
end
end

if pyramid == true
for Q = 1

Mlength = 600;

idx1  = find(Datap1_1(:,1)>0 & Datap1_1(:,1)<Mlength);
idx2  = find(Datap1_2(:,1)>0 & Datap1_2(:,1)<Mlength);
idx3  = find(Datap1_3(:,1)>0 & Datap1_3(:,1)<Mlength);
```

```
idx4  = find(Datap2_1(:,1)>0 & Datap2_1(:,1)<Mlength);
idx5  = find(Datap2_2(:,1)>0 & Datap2_2(:,1)<Mlength);
idx6  = find(Datap2_3(:,1)>0 & Datap2_3(:,1)<Mlength);

idx7  = find(Datap3_1(:,1)>0 & Datap3_1(:,1)<Mlength);
idx8  = find(Datap3_2(:,1)>0 & Datap3_2(:,1)<Mlength);
idx9  = find(Datap3_3(:,1)>0 & Datap3_3(:,1)<Mlength);


Datap1_1a(:,1) = Datap1_1(idx1,1);
Datap1_1a(:,2) = Datap1_1(idx1,2);
Datap1_2a(:,1) = Datap1_2(idx2,1);
Datap1_2a(:,2) = Datap1_2(idx2,2);
Datap1_3a(:,1) = Datap1_3(idx3,1);
Datap1_3a(:,2) = Datap1_3(idx3,2);
Datap2_1a(:,1) = Datap2_1(idx4,1);
Datap2_1a(:,2) = Datap2_1(idx4,2);
Datap2_2a(:,1) = Datap2_2(idx5,1);
Datap2_2a(:,2) = Datap2_2(idx5,2);
Datap2_3a(:,1) = Datap2_3(idx6,1);
Datap2_3a(:,2) = Datap2_3(idx6,2);
Datap3_1a(:,1) = Datap3_1(idx7,1);
Datap3_1a(:,2) = Datap3_1(idx7,2);
Datap3_2a(:,1) = Datap3_2(idx8,1);
Datap3_2a(:,2) = Datap3_2(idx8,2);
Datap3_3a(:,1) = Datap3_3(idx9,1);
Datap3_3a(:,2) = Datap3_3(idx9,2);


% plot(DataX1_1a(:,1),DataX1_1a(:,2),DataX1_2a(:,1),DataX1_2a(:,2),...
%     DataX1_3a(:,1),DataX1_3a(:,2),DataX1_4a(:,1),DataX1_4a(:,2))


%circle fit
s1 = 300;
s2 = 600;
s3 = 300;
x1(1) = Datap1_3a(s1,1);
x1(2) = Datap1_3a(end-s3,1);
x1(3) = Datap1_3a(s2,1);
y1(1) = Datap1_3a(s1,2);
y1(2) = Datap1_3a(end-s3,2);
y1(3) = Datap1_3a(s2,2);

x2(1) = Datap2_3a(s1,1);
x2(2) = Datap2_3a(end-s3,1);
x2(3) = Datap2_3a(s2,1);
y2(1) = Datap2_3a(s1,2);
y2(2) = Datap2_3a(end-s3,2);
y2(3) = Datap2_3a(s2,2);

x3(1) = Datap3_3a(s1,1);
x3(2) = Datap3_3a(end-s3,1);
x3(3) = Datap3_3a(s2,1);
y3(1) = Datap3_3a(s1,2);
y3(2) = Datap3_3a(end-s3,2);
y3(3) = Datap3_3a(s2,2);

 [R,xcyc] = fit_circle_through_3_points([x1' y1' x2' y2' x3' y3']);
yvals1 = xcyc(2,1)+sqrt(R(1)^2-(Datap1_3a(:,1)-xcyc(1,1)).^2);
yvals2 = xcyc(2,2)+sqrt(R(2)^2-(Datap2_3a(:,1)-xcyc(1,2)).^2);
yvals3 = xcyc(2,3)+sqrt(R(3)^2-(Datap3_3a(:,1)-xcyc(1,3)).^2);

if abs(yvals1(1)-y1(1))>100
    yvals1 =  xcyc(2,1)-sqrt(R(1)^2-(Datap1_3a(:,1)-xcyc(1,1)).^2);
end
if abs(yvals2(1)-y2(1))>100
    yvals2 =  xcyc(2,2)-sqrt(R(2)^2-(Datap2_3a(:,1)-xcyc(1,2)).^2);
end
if abs(yvals3(1)-y3(1))>100
```

```
    yvals3 =  xcyc(2,3)-sqrt(R(3)^2-(Datap3_3a(:,1)-xcyc(1,3)).^2);
end

figure(n)
hold on
plot(Datap1_3a(:,1),Datap1_3a(:,2),Datap1_3a(:,1),yvals1)
plot(Datap2_3a(:,1),Datap2_3a(:,2),Datap2_3a(:,1),yvals2)
plot(Datap3_3a(:,1),Datap3_3a(:,2),Datap3_3a(:,1),yvals3)
legend('1','c1','2','c2','3','c3')
hold off
n=n+1;

%difference of circle fit and original
cdiff1 = Datap1_3a(:,2) - yvals1;
cdiff2 = Datap2_3a(:,2) - yvals2;
cdiff3 = Datap3_3a(:,2) - yvals3;

%drawing target line
refx1 = 0:140;
refx2 = 141:145;
refx3 = 145:445;
refx4 = 445:448;
refx5 = 449:600;

refy1 = 0*ones(1,length(refx1));
refy2 = -refx2+refx2(1);
refy3 = -4*ones(1,length(refx3));
refy4 = refx4-refx4(end);
refy5 = 0*ones(1,length(refx5));

refx = [refx1 refx2 refx3 refx4 refx5];
refy = [refy1 refy2 refy3 refy4 refy5];

figure(n)
hold on
plot(Datap1_3a(:,1),cdiff1)
plot(Datap2_3a(:,1),cdiff2)
plot(Datap3_3a(:,1),cdiff3)
plot(refx,refy,'--')
legend('Trial1','Trial2','Trial3','Target')
title('Contours of Pyramid tool (Unloaded, 4mm indent)')
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n = n+1;

%difference of loaded and original

dx = 0:0.1:Mlength;
dy1 = zeros(1,length(dx));
dy2 = zeros(1,length(dx));
dy3 = zeros(1,length(dx));
for k = 1:length(dx)
    tempidx1 = Datap1_1a(:,1) > dx(k)-0.25 & Datap1_1a(:,1) < dx(k)+0.25;
    tempidx2 = Datap1_2a(:,1) > dx(k)-0.25 & Datap1_2a(:,1) < dx(k)+0.25;

    tempidx3 = Datap2_1a(:,1) > dx(k)-0.25 & Datap2_1a(:,1) < dx(k)+0.25;
    tempidx4 = Datap2_2a(:,1) > dx(k)-0.25 & Datap2_2a(:,1) < dx(k)+0.25;

    tempidx5 = Datap3_1a(:,1) > dx(k)-0.25 & Datap3_1a(:,1) < dx(k)+0.25;
    tempidx6 = Datap3_2a(:,1) > dx(k)-0.25 & Datap3_2a(:,1) < dx(k)+0.25;

    dy1(k) = median(Datap1_2a((tempidx2),2))-median(Datap1_1a((tempidx1),2));
    dy2(k) = median(Datap2_2a((tempidx4),2))-median(Datap2_1a((tempidx3),2));
    dy3(k) = median(Datap3_2a((tempidx6),2))-median(Datap3_1a((tempidx5),2));
end
dx1 = 0:0.1:Mlength;
dx2 = 0:0.1:Mlength;
dx3 = 0:0.1:Mlength;

dx1(find(isnan(dy1)))= [];
```

```
dy1(find(isnan(dy1)))= [];
dx2(find(isnan(dy2)))= [];
dy2(find(isnan(dy2)))= [];
dx3(find(isnan(dy3)))= [];
dy3(find(isnan(dy3)))= [];

figure(n)
hold on
plot(dx1,dy1)
plot(dx2,dy2)
plot(dx3,dy3)
plot(refx,refy,'--')
legend('Trial1','Trial2',...
    'Trial3','Target')
title('Contous of Pyramid tool (Loaded, 4mm indent)')
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n = n+1;

%unforced and forced difference
figure(n)
hold on
plot(Datap1_3a(:,1),cdiff1)
plot(Datap2_3a(:,1),cdiff2)
plot(Datap3_3a(:,1),cdiff3)
plot(dx1,dy1)
plot(dx2,dy2)
plot(dx3,dy3)
plot(refx,refy,'--')
legend('Unloaded Pyramid trial1','Unloaded Pyramid trial2',...
    'Unloaded Pyramid trial3'...
   ,'Loaded Pyramid trial1','Loaded Pyramid trial2',...
    'Loaded Pyramid trial3','Target')
title('Contours of Pyramid tool (4mm indent)')
xlabel('Position (mm)')
ylabel('Groove Depth (mm)')
hold off
n = n+1;

glength = 300;
st = 145;
dxa = st:0.2:(st+glength);
dy4 = zeros(1,length(dxa));
dy5 = zeros(1,length(dxa));
dy6 = zeros(1,length(dxa));
dy7 = zeros(1,length(dxa));
dy8 = zeros(1,length(dxa));
dy9 = zeros(1,length(dxa));

dx1 = dx1';
dx2 = dx2';
dx3 = dx3';
dy1 = dy1';
dy2 = dy2';
dy3 = dy3';
for k = 1:length(dxa)
    tempidx7 = dx1 > dxa(k)-0.5 & dx1 < dxa(k)+0.5;
    tempidx8 = Datap1_3a(:,1) > dxa(k)-0.5 & Datap1_3a(:,1) < dxa(k)+0.5;

    tempidx9 = dx2 > dxa(k)-0.5 & dx2 < dxa(k)+0.5;
    tempidx10 = Datap2_3a(:,1) > dxa(k)-0.5 & Datap2_3a(:,1) < dxa(k)+0.5;

    tempidx11 = dx3 > dxa(k)-0.5 & dx3 < dxa(k)+0.5;
    tempidx12 = Datap3_3a(:,1) > dxa(k)-0.5 & Datap3_3a(:,1) < dxa(k)+0.5;

    dy4(k) = (median(cdiff1(tempidx8)) - median(dy1(tempidx7)));
    dy5(k) = (median(cdiff2(tempidx10))- median(dy2(tempidx9)));
    dy6(k) = (median(cdiff3(tempidx12))- median(dy3(tempidx11)));
    dy7(k) = (-4 - median(cdiff1(tempidx8)))/-4*100;
    dy8(k) = (-4 - median(cdiff2(tempidx10)))/-4*100;
```

```
    dy9(k) = (-4 - median(cdiff3(tempidx12)))/-4*100;
end

dx4 = st:0.2:(st+glength);
dx5 = st:0.2:(st+glength);
dx6 = st:0.2:(st+glength);

dx7 = st:0.2:(st+glength);
dx8 = st:0.2:(st+glength);
dx9 = st:0.2:(st+glength);

dx4(find(isnan(dy4)))= [];
dy4(find(isnan(dy4)))= [];
dx5(find(isnan(dy5)))= [];
dy5(find(isnan(dy5)))= [];
dx6(find(isnan(dy6)))= [];
dy6(find(isnan(dy6)))= [];

dx7(find(isnan(dy7)))= [];
dy7(find(isnan(dy7)))= [];
dx8(find(isnan(dy8)))= [];
dy8(find(isnan(dy8)))= [];
dx9(find(isnan(dy9)))= [];
dy9(find(isnan(dy9)))= [];
figure(n)
hold on
plot(dx4,dy4)
plot(dx5,dy5)
plot(dx6,dy6)
legend('Trial1','Trial2','Trial3')
title('Springback of Pyramid tool (4mm indent)')
xlabel('Position (mm)')
ylabel('Difference (mm)')
hold off
n = n+1;

figure(n)
hold on
plot(dx7,dy7)
plot(dx8,dy8)
plot(dx9,dy9)
legend('Trial1','Trial2','Trial3')
title('Springback of Pyramid tool (ref 4mm indent)')
xlabel('Position (mm)')
ylabel('Percent Difference (%)')
hold off
n = n+1;
end
end

if pyramid == true
    if hemispherical == true
        if rigid_roller ==true
            for Q = 1

            end
        end
    end
end
```

References

[1] G. Grzancic, C. Becker and N. B. Khalifa, "Basic Analysis of the Incremental Profile Forming Process," *Journal of Manufacturing Science and Engineering*, vol. 138, pp. 1-6, 2016.

[2] X. Xu, Y. Liu, X. Yang and W. Zhang, "A computational model for the normal pressure between the tube and the hydroforming die," in *WASE International Conference in Information Engineering*, Maanshan, China, 2010.

[3] G. Grzancic, "Process Development and Basic Investigations on Incremental Profile Forming," Ph. D Thesis, Technical University of Dortmund, 2018.

[4] J. Allwood, S. Duncan, J. Cao, P. Groche, G. Hirt, B. Kinsey, T. Kuboki, M. Liewald, A. Sterzing and A. Tekkaya, "Closed-loop Control of Product Properties in Metal Forming," *CIRP Annals-Manufacturing Technology*, vol. 65, pp. 573-596, 2016.

[5] K. Srinivasan and E. Tekkaya, "Closed Loop Control of Incremental Profile Forming Process," Proposal to National Science Foundation and Deutsche Forschungsgemeinschaft, 2018.

[6] P. Groche, P. Beiter and M. Henkelmann, "Prediction and Inline Compensation of Springback in Roll Forming of High and Ultra-high Strength Steels," *Product Engineering - Research and Development*, pp. 401-407, 2008.

[7] L. Zhang, M. Zhao, Y. Zou and S. Gao, "A New Surfance Inspection Method of TWBS Based on Active Laser-Triangulation," *World Congress on Intelligent Control and Automation*, pp. 1174-1179, 2008.

[8] "Confocal sensor systems for displacement, distance, position and thickness," Micro-Epsilon, 2018. [Online]. Available: https://www.micro-epsilon.com/displacement-position-sensors/confocal-sensor/. [Accessed 20 November 2018].

[9] "Laser distance sensors for extra long ranges," Micro-Epsilon, 2018. [Online]. Available: https://www.micro-epsilon.com/displacement-position-sensors/laser-distance-sensor/. [Accessed November 2018].

[10] GOM, "ATOS - Industrial 3D Scanning Technology," 2018. [Online]. Available: http://www.gom.com/metrology-systems/atos.html.

[11] K. Khoshelham and S. O. Elberink, "Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications," *Sensors*, vol. 12, pp. 1437-1454, 2012.

[12] scanCONTROL 2900 Quick Reference and Instruction Manual, Dresden-Langebrück, Germany: Micro-Epsilon, 2016.